

VŠB – Technická univerzita Ostrava

Fakulta strojní

Institut dopravy

**Algoritmus pro sledování dodržování normy letového
zatížení letových posádek**

**Algorithm for Flight Crews Workload Rule Compliance
Monitoring**

Student:

Bc. Miroslav Hrdlička

Vedoucí diplomové práce:

doc. Ing. Vladimír Smrž, Ph.D.

Ostrava 2018

Zadání diplomové práce

Student: **Bc. Miroslav Hrdlička**

Studijní program: N2301 Strojní inženýrství

Studijní obor: 2301T003 Dopravní technika a technologie

Specializace: 40 Letecká doprava

Téma: **Algoritmus pro sledování dodržování normy letového zatížení letových
posádek
Algorithm for Flight Crews Workload Rule Compliance Monitoring**

Jazyk vypracování: čeština

Zásady pro vypracování:

Cíl práce: Vytvořit návrh algoritmu pro sledování dodržování normy zatížení letových posádek použitelný pro malého leteckého dopravce včetně výběru aplikačního SW.

Osnova práce:

1. Úvod.
2. Vývoj aktuálních předpisových požadavků v oblasti letového zatížení posádek.
3. Systém plánování letových posádek u malého leteckého dopravce.
4. Návrh algoritmu pro sledování dodržování normy zatížení letového posádek včetně výběru aplikačního SW.
5. Závěr.

Seznam doporučené odborné literatury:

Nařízení komise (EU) č. 965/2012 ve znění pozdějších předpisů / vybrané části
Nařízení Komise (EU) č. 1178/2011 ve znění pozdějších předpisů / část FCL
Zákon o civilním letectví (č. 49/1997 Sb) ve znění pozdějších předpisů

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Ing. Vladimír Smrž, Ph.D.**

Datum zadání: 08.12.2017

Datum odevzdání: 21.05.2018



doc. Ing. Aleš Slíva, Ph.D.
vedoucí katedry



doc. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení

Prohlašuji, že jsem tuto diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce uvedl a jsem všechny použité podklady a literaturu

V Ostravě

.....

Podpis

Prohlašuji, že

- jsem byl seznámen s tím, že na mou diplomovou (bakalářskou) práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo;
- беру на ве́домі, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně, ke své vnitřní potřebě, diplomovou (bakalářskou) práci užít (§ 35 odst. 3);
- souhlasím s tím, že diplomová (bakalářská) práce bude v elektronické podobě archivována v Ústřední knihovně VŠB-TUO a jeden výtisk bude uložen u vedoucího diplomové (bakalářské) práce. Souhlasím s tím, že bibliografické údaje o diplomové (bakalářské) práci budou zveřejněny v informačním systému VŠB-TUO;
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona;
- bylo sjednáno, že užít své dílo, diplomovou (bakalářskou) práci, nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на ве́домі, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě dne

.....
jméno a příjmení studenta

Bc. Miroslav Hrdlička

Pivovarská 5/7, Bílovec, 743 01, Česká Republika

Poděkování

Můj srdečný dík patří vedoucímu bakalářské práce panu doc. Ing. Vladimír Smrž, Ph.D. za poskytnutí jeho cenných rad odborné vedení a pomoc při tvorbě této práce.

Anotace

HRDLIČKA, M. *Algoritmus pro sledování dodržování normy letového zatížení letových posádek: diplomová práce*. Ostrava: VŠB-Technická univerzita Ostrava, Fakulta strojní, Institut dopravy, 2018, 61 s. Vedoucí práce: Smrž, V

Předmětem diplomové práce je vytvořit algoritmus, jenž by byl schopen sledovat, zdali se dodržuje norma letového zatížení posádek. První část práce se zaměřuje na představení samotné normy letového zatížení posádek. Dále jsou zmíněny zásady pro vytváření algoritmu a je vytvořena modelová situace. Tato modelová situace je použita pro demonstraci správného fungování programu, jenž byl vytvořen na základě zpracovaného algoritmu v softwaru Microsoft Excel.

Klíčová slova: algoritmus, doba služby, doba letové služby, doba odpočinku

Annotation

HRDLIČKA, M. *Algorithm for Flight Crew Workload Rule Compliance Monitoring: Master Thesis*. Ostrava: VŠB – Technical university of Ostrava, Faculty of Mechanical Engineering, Institute of Transport, 2018, 61 p. Thesis head: Smrž V.

The goal of this thesis is to create algorithm, which is capable to monitor, whether flight crew comply with Flight Crew Workload Rules. The first part focuses on presenting all the Flight Crew Workload Rules. Subsequently there are presented some rules about how to make an algorithm. There is also created a model situation to demonstrate proper functioning of a program that was created according to the algorithm. This program was made in the Microsoft Excel software.

Key words: algorithm, duty period, flight duty period, rest period

Obsah

Seznam použitých obrázků.....	10
Seznam tabulek	11
Seznam příloh.....	11
Seznam použitých zkratk.....	12
0 Úvod.....	13
1 Letové zatížení posádek	14
1.1 Aklimatizace	15
1.2 Doba služby	16
1.3 Doba letové služby.....	17
1.4 Doba odpočinku	21
2 Možnosti řešení pomocí algoritmu.....	23
2.1 Náležitosti správného algoritmu	23
2.2 Vývojový diagram	24
2.3 Tvorba algoritmů	26
2.3.1 Sekvence.....	26
2.3.2 Větvení	26
2.3.3 Cykly	30
2.4 Algoritmizace.....	32
3 Řešení algoritmu sledování normy letového zatížení posádek	34
3.1 Určení stupně aklimatizace	35
3.2 Výpočet maximální doby letové služby.....	35

3.3	Prodloužení letové služby	38
3.4	Zkrácení letové služby	40
3.5	Dělená služba	40
3.6	Kumulativní doba letové služby	41
3.7	Výpočet doby služby	42
3.8	Kumulativní doba služby	43
3.9	Výpočet doby odpočinku	44
3.10	Rušivý rozvrh.....	45
3.11	Prodloužený odpočinek.....	46
3.12	Rotace	48
3.13	Konec algoritmu	49
4	Ukázka vytvořeného řešení	50
4.1	Představení programu	50
4.2	Uživatelské rozhraní	51
4.3	Modelová situace	56
4.4	Ukázka fungování programu	57
5	Závěr	59
6	Bibliografie	60

Seznam použitých obrázků

<i>Obrázek č. 1: příklad sekvenčního algoritmu</i>	<i>26</i>
<i>Obrázek č. 2: různé druhy zápisů větvení</i>	<i>27</i>
<i>Obrázek č. 3: zápis spojovacího operátoru &&</i>	<i>28</i>
<i>Obrázek č. 4: zápis spojovacího operátoru </i>	<i>28</i>
<i>Obrázek č. 5: podmínka s úplnou alternativou</i>	<i>29</i>
<i>Obrázek č. 6: podmínka s neúplnou autoritou</i>	<i>29</i>
<i>Obrázek č. 7: větvení s několikanásobnou alternativou</i>	<i>30</i>
<i>Obrázek č. 8: cyklus s daným počtem opakování</i>	<i>30</i>
<i>Obrázek č. 9: cyklus s podmínkou na začátku</i>	<i>31</i>
<i>Obrázek č. 10: cyklus s podmínkou na konci</i>	<i>32</i>
<i>Obrázek č. 11: podprogram Matice Aklimatizace</i>	<i>35</i>
<i>Obrázek č. 12: Podprogram pilot je v neznámém stavu aklimatizace (s i bez FRM) ...</i>	<i>36</i>
<i>Obrázek č. 13: podprogram matice FDP</i>	<i>37</i>
<i>Obrázek č. 14: podprogram FDP s prodloužením bez odpočinku za letu</i>	<i>38</i>
<i>Obrázek č. 15: podprogram Dělená služba</i>	<i>41</i>
<i>Obrázek č. 16: podprogram rušivý rozvrh</i>	<i>46</i>
<i>Obrázek č. 17: podprogram Rotace</i>	<i>48</i>
<i>Obrázek č. 19: vizuální podoba programu</i>	<i>50</i>
<i>Obrázek č. 20: okno k zadání vstupních dat</i>	<i>52</i>
<i>Obrázek č. 21: okno Split Duty</i>	<i>53</i>

<i>Obrázek č. 22: okno prodloužení FDP s odpočinkem za letu</i>	<i>54</i>
<i>Obrázek č. 23: okno Correction</i>	<i>54</i>
<i>Obrázek č. 24: okno Určení stupně aklimatizace.....</i>	<i>55</i>

Seznam tabulek

<i>Tabulka 1: Aklimatizace člena posádky</i>	<i>15</i>
<i>Tabulka 2: Maximální denní FDP pro aklimatizovaného člena posádky.....</i>	<i>17</i>
<i>Tabulka 3: Maximální denní FDP pro neaklimatizovaného člena posádky, bez implementace FRM</i>	<i>18</i>
<i>Tabulka 4: Maximální FDP, kdy provozovatel implementoval FRM</i>	<i>18</i>
<i>Tabulka 5: Maximální denní FDP s prodloužením</i>	<i>19</i>
<i>Tabulka 7: Minimální počet místních nocí v odpočinku pro kompenzaci rozdílů časových pásem.....</i>	<i>22</i>

Seznam příloh

<i>Příloha A: Algoritmus pro sledování dodržování normy letového zatížení posádek</i>
<i>Příloha B: Program pro sledování dodržování normy letového zatížení posádek</i>

Seznam použitých zkratk

CS	Certification Specification	certifikační specifikace
EASA	European Aviation Safety Agency	Evropská agentura pro bezpečnost v letectví
EDDC	Dresden Airport	Letiště Drážďany
ESMS	Malmö Airport	Letiště Malmö
ETA	Estimated Time of Arrival	Předpokládaný čas příletu
ETD	Estimated Time of Departure	Předpokládaný čas odletu
EVRA	Riga International Airport	Mezinárodní letiště Riga
FDP	Flight Duty Period	doba letové služby
FTL	Flight Time Limitations	omezení doby letové služby
FRM	Fatigue Risk Management	Systém řízení rizik spojených s únavou
ICAO	International Civil Aviation Organization	Mezinárodní organizace pro civilní letectví
LD	Local Day	místní den
LKMT	Leoš Janáček Ostrava Airport	Letiště Leoše Janáčka Ostrava
LKTB	Brno-Tuřany Airport	Mezinárodní letiště Brno-Tuřany
LSGG	Geneva Airport	Letiště Ženeva
LN	Local Night	místní noc
ORO	Operators Responsibilities	povinnosti provozovatele
RT	Reference Time	referenční čas
UTC	Coordinated Universal Time	koordinovaný světový čas
WOCL	Window of Circadian Low	útlumová fáze cirkadiánního rytmu

0 Úvod

Sledování dodržování normy letového zatížení posádek je nelehký úkol, pokud se jej člen posádky snaží provádět sám. Problémem není náročnost výpočtu, jako spíše množství dat, procesů a proměnných, které musí brát v úvahu. Z tohoto důvodu je vytvořen algoritmus, který všechny procesy a proměnné bere v potaz. O výpočet velkého počtu dat se potom postará program, který je vytvořen podle zmíněného algoritmu.

Produktů pro sledování normy letového zatížení posádek existuje komerčně celá řada. Zpravidla to jsou zpoplatněné programy určené pro velké letecké společnosti, které musí sofistikovaně plánovat posádky. Tento algoritmus a k němu vytvořený program se ale zaměřuje právě na malé společnosti, kde počet pilotů představují spíše jednociferné číslo. Zároveň jsou potřeby tohoto algoritmu čistě akademické. Přes to všechno podléhá algoritmus stejným pravidlům jako již zmíněné komerční produkty.

Nejdříve ze všeho se musí zjistit, co a jak počítat. Všechny informace jsou obsaženy v předpise Nařízení komise (EU) č. 83/2014 známém též jako Part ORO hlava FTL a jeho doplňujících dokumentech. Proto je jim v práci přikládána velká pozornost.

Algoritmus musí splňovat určité zásady, které jsou zde rovněž popsány. Pro nejlepší a nejnázornější zobrazení algoritmu byla vybrána metoda vývojových diagramů, především pro její přehlednost. S využitím získaných informací se následně vytvoří složitý a sofistikovaný algoritmus.

U nově vzniklého algoritmu je potřeba ještě otestovat jeho funkčnost a správnost. K tomu poslouží převedení algoritmu do programátorského rozhraní Visual Basic v softwaru Microsoft Excel. Program se naprogramuje přesně podle předlohy algoritmu a bude si tak možno vyzkoušet celá komplexnost algoritmu.

1 Letové zatížení posádek

Posádky, jinými slovy piloti a palubní průvodčí, jsou vystavováni neustálému pracovnímu zatížení. [1] Aby nedocházelo k přetížení posádek, byla vytvořena nařízení, které mají za úkol regulovat jejich pracovní dobu. Státy Evropské unie (popřípadě jiné státy Evropy) se zavázaly dodržovat nařízení Evropské rady. Konkrétně se jedná o předpis Part ORO hlava FTL [2].

Zmíněný předpis garantuje všem posádkám působící v obchodní letecké dopravě základní práva a zároveň leteckým provozovatelům uděluje povinnosti v oblasti letového zatížení posádek. Pro provozovatele zabezpečující obchodní leteckou přepravu ať už s pravidelnými nebo charterovými linkami platí další povinnosti. Ty jsou popsány jako certifikační specifikace, které obsahuje dokument CS-FTL.1 [2].

Pro posuzování letového zatížení posádek slouží zejména tři hodnoty. Doba služby je jednou z nich. Službou se rozumí jakýkoliv úkol od provozovatele, který může zahrnovat převážně letovou službu, ale také přemísťování, letovou zálohu aj. Čas, který člen posádky stráví ve službě, se nazývá doba služby. Ta začíná od doby, kdy je člen posádky u provozovatele vyžadován k výkonu služby a končí v době, kdy po něm již není žádná další služba vyžadována. [3]

Letová služba, je specifický typ služby, které předpis věnuje největší pozornost. Pomocí letové služby (FDP) přichází provozovatel k zisku, a proto je pro něj nejvýhodnější, když je co nejdelší. FDP začíná ohlášením pilota ke službě, jež zahrnuje jeden nebo více sektorů (část letové služby počínaje od zahájení pojíždění se vzletem a konče uvedením letounu do klidu na požadované parkovací ploše). Letová služba končí při uvedení letounu do klidu během posledního sektoru. [3]

Poslední ze tří hodnot je doba odpočinku. Během této periody je pilot oprostěn od jakékoliv služby. Odpočinek je kontinuální a musí vždy předcházet jakékoliv službě. [3]

1.1 Aklimatizace

Letecká doprava je takový druh odvětví, jehož činností piloti překonávají velké vzdálenosti. To sebou nese řadu problémů a tím největším je rozdíl časových pásem. Tyto časové rozdíly mohou přispívat k únavě posádek.

Aklimatizovaný člen posádky je v takovém stavu, kdy se jeho cirkadiánní hodiny synchronizují s tou časovou zónou, kde se právě nachází. Člen posádky se považuje automaticky za aklimatizovaného na časová pásma v rozsahu ± 2 hodin od časového pásma místa vzletu. Pro časová pásma ± 4 hodiny a více se již pro členy posádky musí určit stupeň aklimatizace. K těmto účelům je zde následující tabulka. [3]:

Tabulka 1: Aklimatizace člena posádky

Rozdíl mezi referenčním časem a místním časem, kde člen posádky zahajuje další službu (h)	Doba, která uběhla od hlášení do služby v referenčním čase				
	< 48	48 - 71:59	72 - 95:59	96 - 119:59	≥ 120
< 4	B	D	D	D	D
≤ 6	B	X	D	D	D
≤ 9	B	X	X	D	D
≤ 12	B	X	X	X	D

Výsledný stupeň aklimatizace může nabývat třech různých hodnot. V případě „B“ neuběhlo od konce poslední služby tak moc času a pilot je proto aklimatizován k času místa vzletu poslední letové služby. [3]

Výsledek „D“ znamená, že je pilot aklimatizován k času, kde začíná další služba. Pilot měl tedy dostatek času, aby se na danou časovou zónu zcela aklimatizoval. [3]

Poslední hodnota „X“ značí, že je pilot v neznámém stavu aklimatizace. Z toho plynou určitá omezení, co se týče výpočtu maximální doby letové služby. Pilot si totiž nestačil dostatečně navyknout na časové pásmo, ve kterém se nachází. [3]

1.2 Doba služby

Doba služby (DP) jako taková nemá v předpise definovány žádné konkrétní limity. Ty jsou jen pro některé konkrétní druhy služeb. Co se ale kontroluje, tak to jsou kumulativní doby služby. Cílem je zabránit tomu, aby se do určitého časového intervalu nenahromadil velký počet služeb. Celkový součet dob služeb nesmí přesáhnout [3]:

- 60 hodin během 7 po sobě jdoucích dní
- 110 hodin během 14 po sobě jdoucích dní
- 190 hodin během 28 po sobě jdoucích dní

Jedním z limitujících faktorů pro čas, kdy naplánovat další DP je vznik rušivého rozvrhu. Tako vytvořený rozvrh služeb narušuje možnost optimálního spánku. [3]

Rozvrh služeb se považuje za rušivý, pokud služba zasahuje do časného (brzkého) začátku, pozdního konce služby nebo do noční služby. [3]

Časný typ:

- Doba služby „začíná brzy,“ pokud nastane v době mezi 05:00 až 05:59 hodin v časovém pásmu, na které je člen posádky aklimatizován.
- Doba služby „končí pozdě“, pokud skončí v době mezi 23:00 až 1:59 hodin v časovém pásmu, na které je člen posádky aklimatizován.

Pozdní typ:

- Doba služby „začíná brzy,“ pokud nastane v době mezi 05:00 až 06:59 hodin v časovém pásmu, na které je člen posádky aklimatizován.
- Doba služby „končí pozdě,“ pokud skončí v době mezi 00:00 až 01:59 hodin v časovém pásmu, na které je člen posádky aklimatizován.

Noční služba: DP se považuje za noční službu, pokud je její součástí nebo zasahuje do časového intervalu mezi 2:00 až 4:59 hodin. [3]

1.3 Doba letové služby

Doba letové služby nesmí přesáhnout maximální povolenou FDP. Tuto hodnotu lze získat z tabulky 2. Jedná se o základní maximální dobu letové služby bez žádného prodloužení. Výsledná hodnota závisí na tom, kdy letová služba započala. Začátek FDP se odvíjí od referenčního času. Tím se rozumí místní čas v rozsahu ± 2 hodiny od místního času, na který je pilot aklimatizovaný. Druhým parametrem je počet sektorů s limitem 10 sektorů na jednu letovou službu. [3]

Tabulka 2: Maximální denní FDP pro aklimatizovaného člena posádky

Začátek FDP v referenčním čase	Počet sektorů								
	1 – 2	3	4	5	6	7	8	9	10
06:00 – 13:29	13:00	12:30	12:00	11:30	11:00	10:30	10:00	9:30	9:00
13:30 – 13:59	12:45	12:15	11:45	11:15	10:45	10:15	9:45	9:15	9:00
14:00 – 14:29	12:30	12:00	11:30	11:00	10:30	10:00	9:30	9:00	9:00
14:30 – 14:59	12:15	11:45	11:15	10:45	10:15	9:45	9:15	9:00	9:00
15:00 – 15:29	12:00	11:30	11:00	10:30	10:00	9:30	9:00	9:00	9:00
15:30 – 15:59	11:45	11:15	10:45	10:15	9:45	9:15	9:00	9:00	9:00
16:00 – 16:29	11:30	11:00	10:30	10:00	9:30	9:00	9:00	9:00	9:00
16:30 – 16:59	11:15	10:45	10:15	9:45	9:15	9:00	9:00	9:00	9:00
17:00 – 04:59	11:00	10:30	10:00	9:30	9:00	9:00	9:00	9:00	9:00
05:00 – 05:14	12:00	11:30	11:00	10:30	10:00	9:30	9:00	9:00	9:00
05:15 – 05:29	12:15	11:45	11:15	10:45	10:15	9:45	9:15	9:00	9:00
05:30 – 05:44	12:30	12:00	11:30	11:00	10:30	10:00	9:30	9:00	9:00
05:45 – 05:59	12:45	12:15	11:45	11:15	10:45	10:15	9:45	9:15	9:00

Tabulka 2 je platná pro případy, kdy je pilot aklimatizován na místo k času místa vzletu poslední služby nebo k místu vzletu další služby. Pro situace, kdy je pilot v neznámém stavu aklimatizace, slouží jiná tabulka.

Jedná se o tabulku 3. Zde již nezáleží na začátku FDP, ale pouze na počtu sektorů. Maximální počet sektorů se snižuje na 8. [3]

Tabulka 3: Maximální denní FDP pro neaklimatizovaného člena posádky, bez implementace FRM

Počet sektorů								
1 – 2	3	4	5	6	7	8	9	10
11:00	10:30	10:00	9:30	9:00	9:00	9:00		

Maximální dobu letové služby, kdy je pilot v neznámém stavu aklimatizace lze ale pro některé sektory ještě prodloužit. Operátor v tomto případě musí implementovat metodiku FRM. Jedná se o systém rizik spojených s únavou, který bývá součástí provozní příručky a podléhá schválení příslušného leteckého úřadu. [3]

Sektory 9 a 10 jsou stále neprovozovatelné. Pro všechny ostatní počty sektorů (krom 8 sektorů) je ale maximální doba letové služby delší. [3]

Tabulka 4: Maximální FDP, kdy provozovatel implementoval FRM

Počet sektorů								
1 – 2	3	4	5	6	7	8	9	10
12:00	11:30	11:00	10:30	10:00	9:30	9:00		

Základní maximální FDP je možno pro aklimatizované členy posádky dále prodloužit. Jednou z možností prodloužení je bez využití odpočinku za letu. V tomto případě může být letová služba delší až o 1 hodinu. [3]

Prodlužování FDP by se nemělo stát běžnou praktikou provozovatele. Proto lze prodloužení využít jen dvakrát během 7 dní. Navíc se musí prodloužit odpočinek před takovou letovou službou a také po ní o 2 hodiny. Nebo stačí zintenzivnit poletový odpočinek na 4 hodiny. [3]

Taktéž se snižuje maximální možný počet sektorů na 5. Velký důraz se také přikládá na útlumovou fázi cirkadiánního rytmu. Jedná se o dobu mezi 2:00 a 5:59 hodin [3], kdy je snížena pilotova výkonnost na minimum.

Limit pěti sektorů tak platí jen pro případy, kdy FDP nijak nezasahuje do WOCL. Povolený počet sektorů se sníží na 4, pokud letová služba zasáhne dvěma hodinami

nebo méně do WOCL a pro více jak dvě hodiny letové služby, které probíhají ve WOCL platí snížení počtu sektorů na maximálně 2. [3]

Jsou také případy, závislé na času začátku FDP, kdy nelze letovou službu prodloužit vůbec. Jedná se zejména o noční lety. Tyto a další výše zmíněné případy jsou zapsány v tabulce 5:

Tabulka 5: Maximální denní FDP s prodloužením

Začátek FDP v místním čase	Počet sektorů			
	1 – 2	3	4	5
06:00 – 06:14				
06:15 – 06:29	13:15	12:45	12:15	11:45
06:30 – 06:44	13:30	13:00	12:30	12:00
06:45 – 06:59	13:45	13:15	12:45	12:15
07:00 – 13:29	14:00	13:30	13:00	12:30
13:30 – 13:59	13:45	13:15	12:45	
14:00 – 14:29	13:30	13:00	12:30	
14:30 – 14:59	13:15	12:45	12:15	
15:00 – 15:29	13:00	12:30	12:00	
15:30 – 15:59	12:45			
16:00 – 16:29	12:30			
16:30 – 16:59	12:15			
17:00 – 17:29	12:00			
17:30 – 17:59	11:45			
18:00 – 18:29	11:30			
18:30 – 18:59	11:15			
19:00 – 03:59	Nelze prodloužit FDP			
04:00 – 04:14				
04:15 – 04:59				
05:00 – 05:14				
05:14 – 05:29				
05:30 – 05:44				
05:45 – 05:59				

Druhým způsobem, jak prodloužit dobu letové služby je s využitím odpočinku za letu. Oba typy prodloužení FDP se ale nesmí nikdy kombinovat v průběhu jedné služby. [3]

Aby mohl pilot využít odpočinek za letu, musí se zesílit posádka, aby jej někdo nahradil. Pro potřeby výpočtu maximální FDP se posádka může zesílit o jednoho nebo dva členy navíc. Důležité také je, jaké jaký typ zařízení určené pro odpočinek za letu se nachází na palubě letounu. [4]

Předpokládá se, že tenhle typ prodloužení se bude využívat spíše na dlouhých letech. I z tohoto důvodu je snížen počet sektorů nanejvýš na 3. [4]

Tříd odpočinkových zařízení je celkem 3. Každé z nich má svou kvalitu hodnoceno od 1 jako nejlepší až po 3 jako nejhorší [4]:

- Třída 1 – jedná se o lůžko umožňující spánek, které je oddělené od prostoru pro cestující. Je tiché a klidné. Lůžko je buďto zcela vodorovné nebo nakloněno ne méně než 80°.
- Třída 2 – jedná se o sedačku, kterou lze sklopit do ze svislé polohy do polohy alespoň 45° a která má oporu pro nohy. Taková sedačka je dostatečně vysoká i široká a je oddělena od prostoru pro cestující minimálně závěsem. Prostor by měl být v dostatečně utlumeném světle a měl by poskytovat alespoň částečnou zvukotěsnost.
- Třída 3 - jedná se o sedačku, jež se dá sklopit alespoň do polohy 40° a je oddělena minimálně závěsem od prostoru pro cestující.

Maximální doba letové služby zesílené posádky se potom odvíjí od velikosti posádky a třídy zařízení určeného pro odpočinek za letu. Pro tříčlennou posádku platí následující omezení [4]:

Bráno v potaz třída zařízení určené pro odpočinek za letu a zesílení posádky o jednoho člena navíc platí [4]:

- Maximální FDP je 16 hodin pro třídu 1
- Maximální FDP je 15 hodin pro třídu 2
- Maximální FDP je 14 hodin pro třídu 3

Pro 2 členy posádky potom platí [4]:

- Maximální FDP je 17 hodin pro třídu 1
- Maximální FDP je 16 hodin pro třídu 2
- Maximální FDP je 15 hodin pro třídu 3

K tomu všemu lze ještě dodatečně prodloužit letovou službu o 1 hodinu za předpokladu, že posádka letí 1 nebo 2 sektory a jeden z nich trvá alespoň 9 hodin. [4]

Existuje ještě jedna možnost, jak prodloužit maximální dobu letové služby, a to za pomoci dělené služby. [3] Dělená služba se bere jako letová služba a zahrnuje přestávku na zemi. Tato přestávka musí být dlouhá alespoň 3 celé hodiny. Až o 50 % času přestávky se může FDP prodloužit. V případě, že je přestávka na zemi delší než 6 hodin nebo zasahuje do WOCL, nesmí se využít k prodloužení FDP. [4]

Stejně jak tomu bylo u doby služby, tak se i u doby letové služby kontroluje kumulativní FDP. Cíl se taktéž nemění. Je snaha předejít nadměrnému přetěžování člena posádky během určitého časového intervalu. Kumulativní FDP nesmí přesáhnout [3]:

- 100 hodin během 28 po sobě jdoucích dní
- 900 hodin během kalendářního roku
- 1000 hodin během 12 po sobě jdoucích měsíců

1.4 Doba odpočinku

Pro určování minimální doby odpočinku, která následuje před či po každé službě záleží na tom, zda je pilot v home base nebo nikoliv. Pojem home base se dá charakterizovat jako mateřské letiště či domácí základna. Jedná se o místo, kde člen posádky obvykle začíná a končí svou službu a provozovatel není zodpovědný za sehnání ubytování. [3]

Minimální doba odpočinku před FDP začínající v home base je alespoň tak dlouhá, jako byla předcházející služba. V případech, kdy byla předešlá DP kratší než dvanáct hodin, je minimální dobou odpočinku právě těchto 12 hodin. [3]

Pro DP, které začíná mimo home base platí, že musí minimální odpočinek být alespoň tak dlouhý jako byla předešlá služba nebo 10 hodin, podle toho, co je větší. Pilotovi musí být umožněno alespoň 8 hodin spánku, z čehož plyne, že nanejvýš 2 hodiny jsou určeny k cestě z letiště k ubytování a zpět. [3]

Odpočinek lze případně i zkrátit. K tomuto kroku může provozovatel přistoupit jen v případě, že implementoval FRM. V rámci dvou opakujících se prodloužených odpočinků je možné zkrátit dobu odpočinku jen dvakrát. [4]

Pilotům musí být dána možnost na zotavení, aby se předešlo kumulativní únavě. Z toho důvodu předpis vyžaduje, aby se v pravidelných intervalech členům posádky umožnil odpočinek alespoň 36 hodin včetně dvou místních nocí. Doba mezi dvěma prodlouženými odpočinky nesmí být delší než 168 hodin. Dvakrát během každého měsíce by se měl odpočinek prodloužit na 2 místní dny. [3]

Aby si mohl pilot dostatečně odpočat, pokud se je naplánovaná služba rušivým rozvrhem, ale pouze v home base, musí doba odpočinku mezi dvěma DP zahrnovat jednu místní noc. Pokud pilot provede 4 nebo více nočních služeb, časných startů nebo pozdních konců služby mezi dvěma prodlouženými odpočinky, doba dalšího prodlouženého odpočinku se zvýší na 60 hodin. [4]

Rotace je řada služeb (včetně letových služeb) a doba odpočinku mimo home base. Začátek a konec rotace je ale v home base. Pokud se u rotace předpokládá rozdíl mezi časovými pásmy více jak 4 hodiny, minimální odpočinek musí zahrnovat počet místních nocí, které jsou zapsány v tabulce 7. [4]

Tabulka 6: Minimální počet místních nocí v odpočinku pro kompenzaci rozdílů časových pásem

Maximální rozdíl časů mezi RT a místní časem, kde člen posádky provádí odpočinek během rotace	Čas, který uběhl od hlášení první FDP rotace zahrnující alespoň 4 hodinový časový rozdíl			
	< 48	48 – 71:59	72 – 95:59	≥ 96
≤ 6	2	2	3	3
> 6 and ≤ 9	2	3	3	4
> 9 and ≤ 12	2	3	4	5

Pokud se mění lety ze západních na východní nebo z východních na západní, je třeba mít v home base odpočinek o délce nejméně třech místních nocí, aby se neprojevovaly příznaky únavy. [4]

2 Možnosti řešení pomocí algoritmu

Termín algoritmus představuje určitý návod či postup, který je přesně definovaný. Algoritmy jsou zpravidla v obecné formě a až zadáním měnitelných vstupních údajů lze dojít k požadovaným výsledkům. [5] Častými případy, kdy se lidé setkávají s algoritmy jsou například počítačové programy nebo matematické vzorce apod.

2.1 Náležitosti správného algoritmu

Definovat však algoritmus pouze jednou větou nestačí. Aby se daný postup mohl nazývat algoritmem, musí splňovat několik podmínek. Tou první je nutnost mít **začátek a konec** algoritmu. Zatímco mít začátek se zdá být poměrně intuitivní, definovat konec algoritmu už tak jasné být nemusí. Pokud by byla tato podmínka porušena, algoritmus by nikdy nemohl skončit a proces by nikdy nemohl dojít k čtenému výsledku. [6]

Dalším předpokladem pro správný algoritmus je jeho **věcná správnost**. Jinými slovy je potřeba dávat pozor, aby se vzorce do algoritmů nepřepsaly nesprávně. Matematické rovnice se mohou do algoritmů přepsat ve stejné grafické podobě (např.: zlomky $x = \frac{a+b}{c+d}$, odmocniny $\sqrt{x+y}$ apod.). Je-li algoritmus vytvořen pro účely programování, je výhodnější již při jeho vývoji používat programátorský zápis [$x = (a+b)/(c+d)$, $SQRT(x+y)$ atd.], jelikož se tím ušetří čas. [6]

Podmínka věcné správnosti algoritmu skýtá jedno veliké nebezpečí. Je-li podmínka porušena, nemusí se chyba na první pohled projevit. Algoritmus pracuje dál, ale poskytuje nesprávné výsledky. [6]

Jednoznačnost algoritmu je nejčastější chybou, kterou se lidé při jeho vytváření dopouští. Tvůrce si musí dobře promyslet, jestli do svého algoritmu zapracoval všechny možnosti, které mohou nastat. Jinými slovy si tvůrce musí uvědomit všechny omezující podmínky. [6]

Jiný způsob, jak porušit pravidlo jednoznačnosti je případ, kdy by se algoritmus větvil bez toho, aby byla předem vytvořena podmínka (viz rozhodovací blok str. 25). Člověk by takto „nejednoznačný“ algoritmus pochopit mohl, stroje ale potřebují přesně definované algoritmy. [6]

Správný algoritmus musí být taktéž **obecný**. Vytvořit jej pouze pro jednu situaci a konkrétní hodnoty by bylo značně nepraktické. Pro danou situaci by algoritmus stačil, ale nedal by se znovu zopakovat pro jiné parametry. Jisté algoritmy navíc mohou sloužit jako podprogramy pro jiné algoritmy. Tvůrce tak nemusí, v případě dostatečně obecných algoritmů, namísto každého podprogramu vytvářet nový algoritmus. [6]

Jedním z předpokladů vyhovujícího algoritmu je podmínka **opakovatelnosti**. Pokaždé, když se dosadí stejné hodnoty, výsledek by měl být vždy stejný (algoritmus se bude chovat pokaždé stejně). Příklad, kdy by byla tato podmínka porušena může nastat, když je algoritmus součástí rozsáhlejšího algoritmu a jedna či více proměnných nejsou definované. Mohou tak nabývat libovolných hodnot. [6]

A nakonec, každý algoritmus by měl být **srozumitelný**. Je potřeba, aby mu rozuměla jednak osoba, které je určen, ale i sám tvůrce. Může se totiž stát, bude nutné algoritmus upravit nebo vylepšit. K tomu, aby byl algoritmus dostatečně srozumitelný, napomáhají metody pro zápis algoritmů. [6]

2.2 Vývojový diagram

Algoritmus lze zapsat vícero způsoby. Každý z nich má své výhody a nevýhody. Mezi nejužívanější metody patří [6]:

- Slovní vyjádření
- Matematický zápis
- Rozhodovací tabulky
- Počítačové programy
- Vývojové diagramy

Poslední zmíněná metoda bude v následujících algoritmech preferována. Výhoda vývojových diagramů je především jejich názornost. Zejména proto se hojně používá při spolupráci analytiků a programátorů. [6]

Tvorba vývojových diagramů podléhá určitým normovaným standardům, které jsou popsány v normě ČSN ISO 5807. K zápisu příkazů jsou použity značky, přičemž každá z nich má svůj význam. Značky mezi sebou propojují tzv. spojnice. Ty se pro přehlednost zakreslují pouze ve vodorovné nebo svislé poloze a mohou být orientované, anebo neorientované. [5] Zde jsou uvedeny nejběžněji používané značky:



Začátek/Konec

Mezní značka získala svůj název právě proto, že vymezují celý algoritmus (popřípadě podprogram). Jinými slovy udává, kde je začátek a kde konec. Z toho důvodu má tato mezní značka vždy jen jednu orientovanou hranu (spojnici). Díky tomu může mít pouze jeden vstup (začátek), nebo jeden výstup (konec). [7]



Vstup/Výstup

Tato značka zabezpečuje operaci s daty ve fázi jejich vstupu či výstupu z algoritmu. [7] Symbol má pouze jeden vstup a jeden výstup, a to proto, že je součástí tzv. sekvenčních bloků. Algoritmus se v tomto případě nesmí rozvětvit. [6]



Zpracování

Symbol zpracování představuje operaci, při které dochází k přeměně dat (jejich transformace). Vstupů do značky může být i více a mohou být realizovaný samostatnými spojnicemi nebo jednou výslednou spojnici, jež vznikla spojením jednotlivých spojníc. Výstup ale musí být vždy jeden. [7]



Rozhodovací blok

Při procesu, který znázorňuje rozhodovací blok dochází k rozvětvení algoritmu. Rozhodnutí, kterou směrnicí se dále vydat nastane až po vyhodnocení podmínek uvnitř značky. Nejčastěji se algoritmus větví na dvě směrnice (+ nebo ano, pokud je splněna podmínka a – nebo ne, pokud není) méně často pak na tři směrnice (pokud se porovnávají hodnoty pomocí větší, menší a rovno: $>$, $<$, $=$). [7]



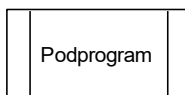
Příprava

Symbol označuje tzv. přípravnou fázi programu. Ta se používá například pro zahájení cyklu o známém počtu opakování. [7]



1

Značka s názvem spojka umožňuje spojit dvě části vývojového diagramu, které nebylo možno nakreslit bez přerušení (konec stránky, křížící se čáry apod.). Spojky, které na konci přerušují a na začátku pokračují musí být označeny stejným číslem nebo písmenem. [7]



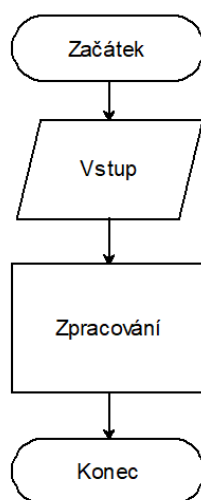
Symbol podprogramu znázorňuje samostatnou část programu, která může obsahovat větší množství kroků. Jedná se o předem definovanou činnost nebo obecnou proceduru. [7]

2.3 Tvorba algoritmů

Při tvorbě algoritmů a s tím souvisejícími vývojovými diagramy se využívá několika postupů. Mezi tyto postupy patří právě sekvence, větvení a cykly.

2.3.1 Sekvence

Algoritmus sloužený pouze ze sekvenčních bloků je nejjednodušším typem algoritmu. Jelikož se nikam nevětví, bývá zpravidla krátký. Jedná se o posloupnost kroků, které se bez skoků zpracovávají jeden za druhým. [8]



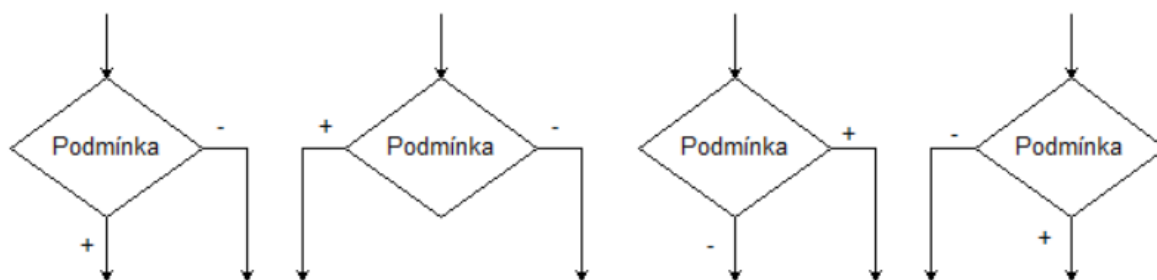
Obrázek č. 1: příklad sekvenčního algoritmu

2.3.2 Větvení

To, co platilo u sekvenčních algoritmů z hlediska počtu výstupů neplatí u větvení. Větvením algoritmus nabírá na komplexnosti, přesto je to jeho nedílnou součástí. Téměř ve všech případech vychází z rozhodovacího bloku dva výstupy (vyjíměčně 3). [8]

Pokud je podmínka splněna (+) vydá se algoritmus jedním směrem a pokud ne (-), vydá se druhým. Protože jsou algoritmy chronologické a nejčastěji postupují shora dolů, přichází i vstup do rozhodovacího bloku většinou shora. Výstupy pak lze různě kombinovat,

jako je tomu na obrázku č. 2. Protože není přímo dané, která pozice větve splňuje podmínku a která nikoliv, je důležité nezapomenout všechny větve správně označit. [8]



Obrázek č. 2: různé druhy zápisů větvení

Pro charakterizování podmínky se často využívá textu. Pokud je to možné, je výhodnější podmínku zapsat pomocí relačních operátorů. Šetří se místem a zápis je přesnější a rychlejší. Výhodou také je, že pokud se podle algoritmu vytváří počítačový program, zápis s relačními operátory se velmi podobá i zápisu v programovacím jazyce. [8]

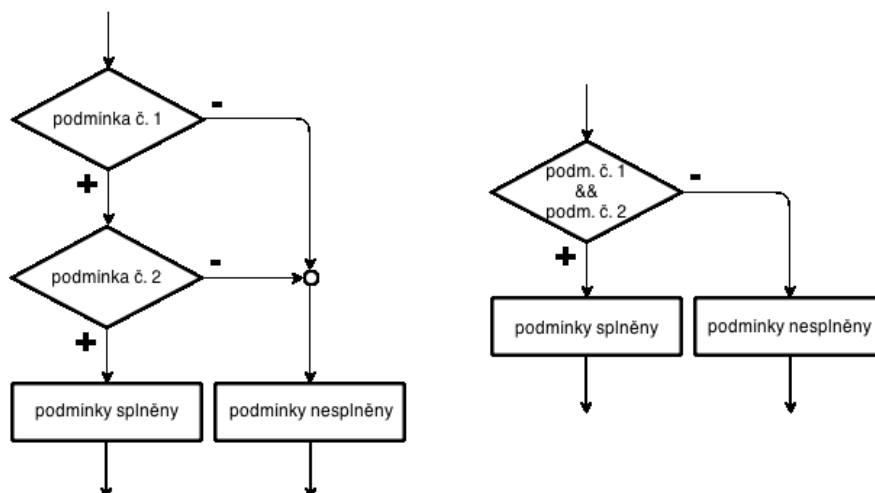
Relační operátory jsou všem známy z matematiky. Jako nejběžnější se používají [8]:

- == rovno ($x == y$; x je rovno y)
- > větší než ($x > y$; x je větší než y)
- < menší než ($x < y$; x je menší než y)
- <> nerovno ($x <> y$; x je různé od y , nebo x není rovno y)
- != nerovno ($x != y$; x je různé od y nebo x není rovno y – jedná se o ekvivalentní zápis předchozího operátoru)
- >= větší nebo rovno ($x >= y$; x je větší nebo rovno y)
- <= menší nebo rovno ($x <= y$; x je menší nebo rovno y)

Obecně platí, že čím je algoritmus delší a složitější, tím více se větví a naopak. Pokud jsou splněny určité podmínky, mohou se některé rozhodovací bloky sloučit do jednoho. Místo toho, aby se ošetřila každá podmínka zvlášť, provede se vše najednou. Rozhodovací bloky se tedy mohou sloučit, pokud je lze zapsat s využitím logických operátorů AND a OR. Někdy se jim také říká spojovací operátory. [8]

Pokud se využije spojovací operátor AND, musí být splněny obě podmínky, které operátor spojuje, aby byla sloučená podmínka vyhodnocena jako splněná (+). Do rozhodovacího bloku se může operátor zapsat jako české „a“, anglické „and“ ale jako nejvýhodnější zápis s přihlédnutím na pozdější zpracování do počítačového programu se jeví varianta „&&.“ [8]

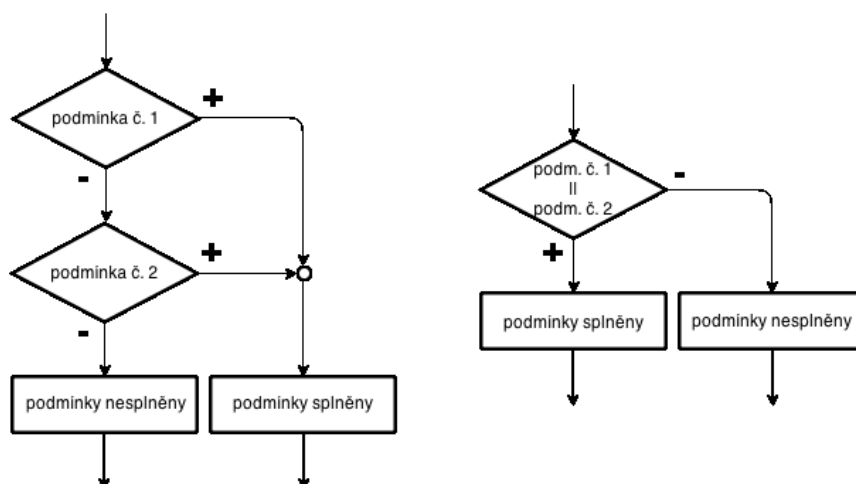
Grafické znázornění operátoru AND je na obrázku č. 3. Je zde jasné vidět úspora místa. Lze také vidět, že jediný způsob, jak u spojené podmínky dosáhnout kladné větve je při splnění všech dílčích podmínek. [8]



Obrázek č. 3: zápis spojovacího operátoru &&

Druhým spojovacím operátorem je OR. Jediný způsob, jak dosáhnout nesplnění spojené podmínky je při nesplnění všech dílčích podmínek. Při splnění byť jen jedné dílčí podmínky, je výsledná větev kladná. [8]

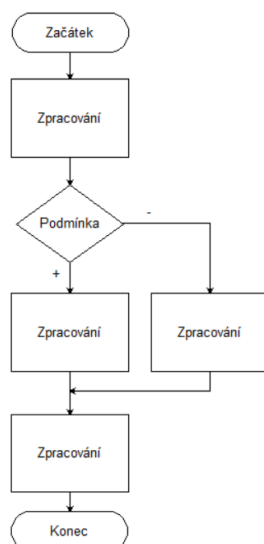
Do vývojových diagramů se operátor zapisuje jako české „nebo,“ anglické „or.“ Opět je ale výhodné zapsat jej jako „||.“ Grafické znázornění operátoru OR je možné shlédnout na obrázku č. 4. [8]



Obrázek č. 4: zápis spojovacího operátoru ||

Úplná alternativa

Větvení se rozděluje podle toho, jaké operace v jednotlivých větvích následují. Jako úplná alternativa se označuje takový typ větvení, kdy se při průchodu obou větví provede nějaký úkon. Když je podmínka splněna, provede se jedna část větve, pokud splněna není, provede se druhá část větve s jiným procesem. [8]



Obrázek č. 5: podmínka s úplnou alternativou

Neúplná alternativa

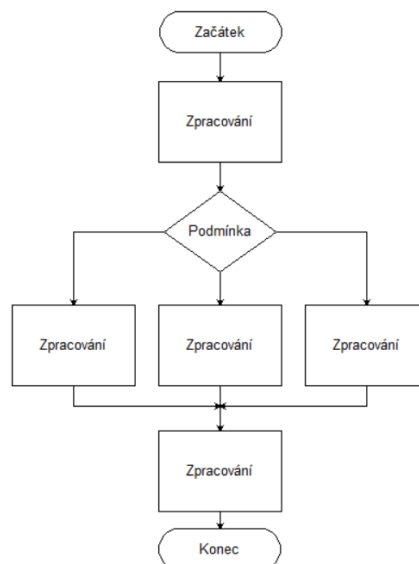
Jak je vidět na obrázku 6, neliší se tento případ příliš od předchozího, jen je jedna jeho větev prázdná. Pokud je podmínka pravdivá, provede se sekvence jednoho či více úkonů. V případě nesplnění podmínky se neprovede žádný úkon a algoritmus pokračuje dál. [8]



Obrázek č. 6: podmínka s neúplnou autoritou

Několikanásobná alternativa

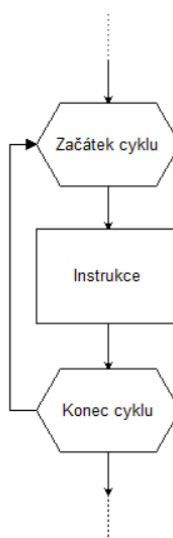
Několikanásobná alternativa je jediný případ, kdy z rozhodovacího bloku vycházejí 3 větve. Tato alternativa se využívá s relačními operátory $<$, $>$ a $=$. [6] Podle toho, čemu se rovná porovnávaná proměnná, takovou ze tří větví bude algoritmus dále pokračovat. [8]



Obrázek č. 7: větvení s několikanásobnou alternativou

2.3.3 Cykly

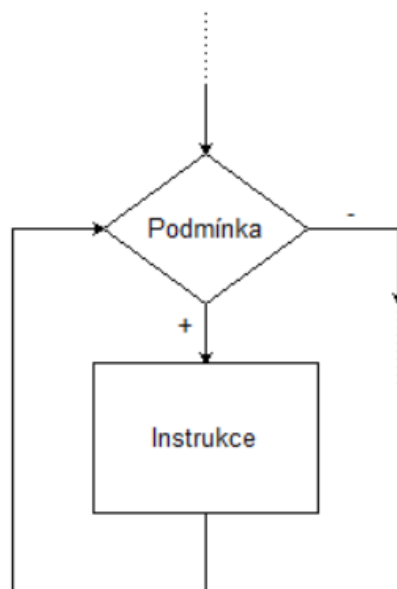
Jak již název napovídá, cyklus je opakující se děj nebo postup v problematice algoritmů. Určujícím faktorem u cyklů je počet opakování. Přesně dané je to u cyklů s pevný počtem opakování. Tento druh cyklu pracuje s tzv. řídicí proměnnou, jejíž hodnota se každým pro každé nové opakování zvýší o předem definovaný přírůstek. Až dosáhne hodnota řídicí proměnné předem stanovené hodnoty, cyklus je ukončen a může se pokračovat dál. [6]



Obrázek č. 8: cyklus s daným počtem opakování

Cyklus s podmínkou na začátku

V případě tohoto cyklu není počet opakování předem znám. Závisí pouze na vstupní podmínce. Může se dokonce stát, že se tělem cyklu neprojde ani jednou. V opačném případě se bude cyklus opakovat tak dlouho, dokud nebude vstupní podmínka splněna. [8]



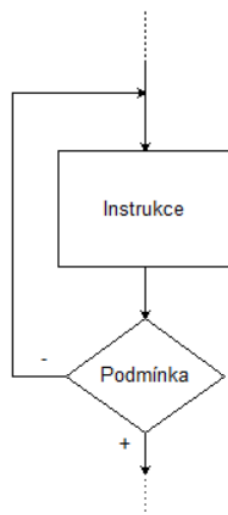
Obrázek č. 9: cyklus s podmínkou na začátku

Používáním tohoto cyklu se tvůrce algoritmu vystavuje určitému riziku. Když se podmínka špatně ošetří nebo se zle navrhnou instrukce v těle cyklu, může se stát, že se bude cyklus opakovat do nekonečna. To není samozřejmě žádaný jev, neboť algoritmus musí být vždy konečný. [8]

Cyklus s podmínkou na konci

Posledním z třech typů cyklů je cyklus s podmínkou na konci. Taktéž zde není předem znám počet opakování a stejně jako předtím i zde počet opakování závisí na podmínce. Liší se jen poloha rozhodovacího bloku, který je umístěn na konci cyklu. [8]

Takové uspořádání zaručuje, že algoritmus tělem cyklu projde minimálně jednou. Dokud se nevyhoví podmínce na konci, cyklus se bude neustále opakovat. Opět hrozí riziko nekonečného opakování cyklu z důvodu špatně nastavených hodnot. [6]



Obrázek č. 10: cyklus s podmínkou na konci

Přece jen ale existují speciální typy algoritmů, kde jsou nekonečné cykly žádané. Takové algoritmy se však objevují zcela výjimečně. Pokud by se za běžných okolností, na základě algoritmus s nekonečným cyklem, vytvořil program, nešel by zastavit a musel by se násilně ukončit. [8]

Obecně lze říci, že výsledek algoritmu musí být znám v konečném počtu kroků. Bez využití cyklů a větvení, by asi nebylo prakticky možno vytvořit plnohodnotný algoritmus. [8]

2.4 Algoritmizace

Tvorbou algoritmu se vytváří určitý obecný postup na danou problematiku. Často se algoritmy používají jako předlohy pro programátorské kódy. Jestliže se algoritmus vytvoří za účelem tvorby počítačového programu, nazývá se tento proces algoritmizací. Algoritmizaci lze rozdělit do několika etap [9]

1. Formulace problému
2. Analýza úlohy
3. Vytvoření algoritmu
4. Sestavení programu
5. Odladění programu

Formulace problému

Cílem první etapy je formulovat, co bude úkolem algoritmu, jakých výsledků by měl dosahovat, jaké jsou vstupní hodnoty a v neposlední řadě také jakých přesností budou výsledky dosahovat. [9]

Analýza úlohy

V tomto kroku se připravovaný algoritmus rozebere více do hloubky. Mělo by se zjistit, jestli je vůbec proveditelný. Měl by vzniknout ucelenější obrázek o struktuře a řešení algoritmu. Proveďte kontrolu, jestli nemá algoritmus více řešení. V případě, že má, tak se vybere to nevhodnější. [9]

Vytvoření algoritmu úlohy

Jednotlivé bloky, ze kterých se algoritmus skládá se sestaví do takového sledu operací, které zaručí žádané výsledky algoritmu. Výsledkem se získá postup, jenž sám o sobě žádné odpovědi nepřináší, ale obsahuje klíč, jak se k nim dopracovat. [9]

Sestavení programu

Nyní se vybere programovací jazyk. Postupy z algoritmu se přenesou do zdrojového kódu programu. Vznikne tak nový program, jehož funkce budou přesně plnit úkoly charakterizované v algoritmu. [9]

Odladění programu

Po sestavení programu je snaha o odstranění všech chyb. Tyto chyby mohou být syntaktické, které vznikají při špatném zápisu kódu nebo nesprávným definováním proměnných. Naštěstí tyto chyby odhalí překladač, do kterého se zdrojový kód zapisuje.

Daleko závažnějšími chybami jsou logické chyby. Důvodem těchto chyb je špatné nebo nedostatečné navržení algoritmu. Program v takovém případě produkuje chybné výsledky nebo funguje nesprávně. Těmto chybám lze nejlépe předejít ve etapě formulace problému nebo analýzy úlohy. [9]

3 Řešení algoritmu sledování normy letového zatížení posádek

Jako vůbec první krok, který se musí udělat pro k tomu, aby se pomocí kroků v algoritmu spočítaly požadované hodnoty je zadání vstupních dat. Prvním z údajů je stupeň aklimatizace pilota, díky čemuž se dál odvíjí způsob výpočtu maximální doby letové služby. Pokud se je pilot jistý, zdali je či není aklimatizovaný na dané místo vzletu, může použít algoritmus k jeho určení anebo jej zadat přímo.

Dalšími údaji jsou začátky a konce doby služby a doby letové služby. Je výhodné tyto časy zadávat v UTC, protože služba může začínat v různých časových pásmech. Jelikož se ale jak doba služby, tak i doba letové služby vypočítává z místního času, musí se čas UTC přepočítat na místní čas tak, že se jako další vstupní údaj přidá odchylka od UTC. Takovéto řešení zadávání času lze rovněž použít pro výpočet stupně aklimatizace.

Poslední údaj, který se musí zadat k výpočtu maximální doby letové služby je počet plánovaných sektorů.

K provedení všech výpočtů ale nestačí pouze znát výše zmíněná vstupní data. Zejména proto, aby bylo možné kontrolovat kumulativní dobu služby a letové služby je potřeba znát předešlé tyto služby. Ty jsou právě uloženy v databázi spolu s předchozími dobami odpočinku. Navíc jsou v databázi nadefinovány hodnoty tabulek, které jsou popsány v kapitole Letové zatížení posádek.

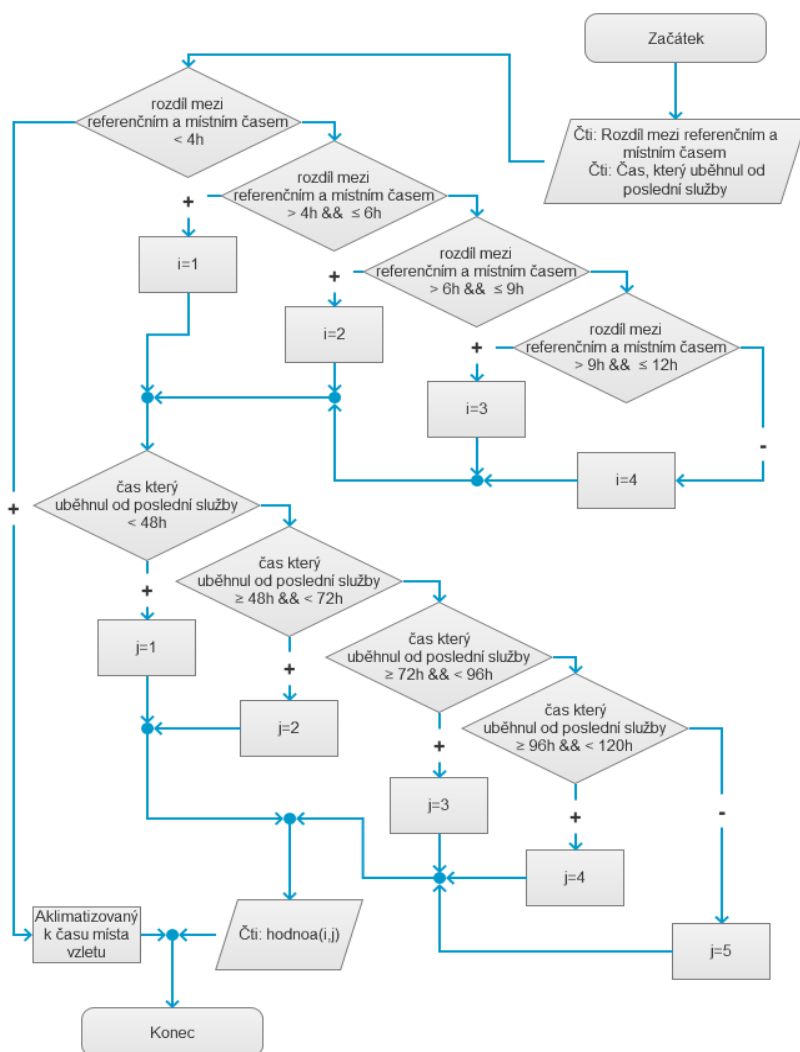
3.1 Určení stupně aklimatizace

Pro výpočet stupně aklimatizace člena posádky se postupuje přesně podle Tabulky 1 (viz výše). V algoritmu se ale podle zmíněné tabulky vytvořila matice $Aklimatizace_{ij}$,

kde i reprezentuje řádek matice a nabývá hodnot: $i=\{1,2,3,4\}$ a j reprezentuje sloupec matice a nabývá hodnot: $j=\{1,2,3,4,5\}$. K určení

hodnoty i je potřeba znát rozdíl mezi referenčním časem a místním časem. Zjednodušeně řečeno se jedná o počet časových pásem od místa ukončení poslední služby k časovému pásmu místa začátku další služby.

Dále se musí zjistit hodnota j , jež je závislá na čase, který uběhnul od poslední služby. Tento čas lze stejně jako rozdíl mezi referenčním a místním časem získat pomocí vstupních hodnot a údajů z databáze předchozích služeb a odpočinků.



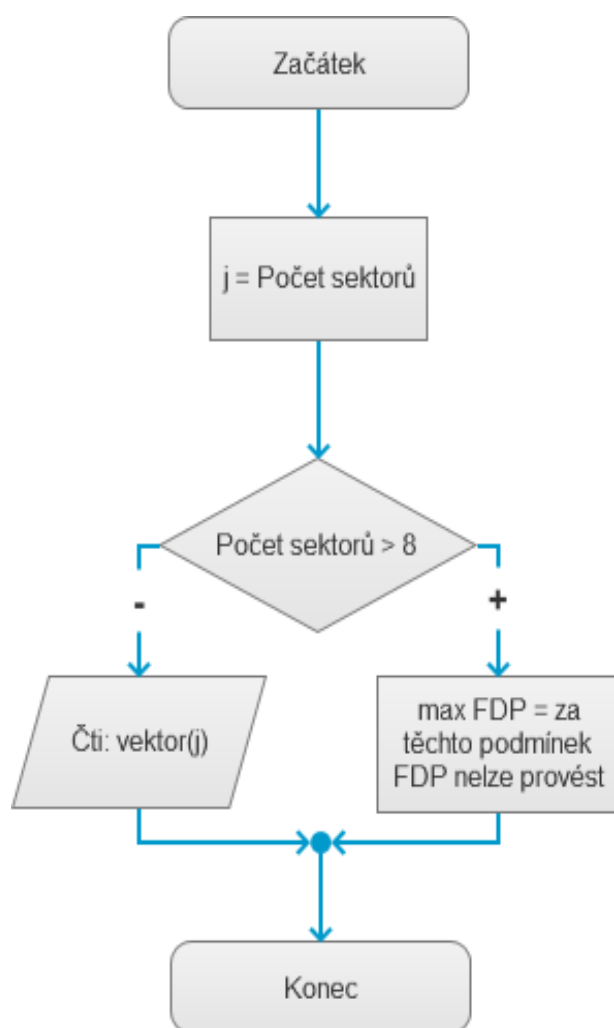
Obrázek č. 11: podprogram Matice Aklimatizace

3.2 Výpočet maximální doby letové služby

Matice $Aklimatizace_{ij}$ nabývá pouze tří hodnot (viz tabulka 1). Pokud je pilot aklimatizován k času místa vzletu nebo k času, kde začíná další jeho služba, pokračuje se s výpočtem maximální doby letové služby pomocí podprogramu matice FDP_{ij} . Je-li však

pilot v neznámém stavu aklimatizace, probíhá výpočet podle podprogramů „Pilot je v neznámém stavu aklimatizace“ a „Pilot je v neznámém stavu aklimatizace pod FRM.“

Nyní je vhodné nejdříve vysvětlit výpočet doby letové služby, když je pilot v neznámém stavu aklimatizace. Maximální FDP nelze v tomto případě již prodloužit. V tomto stupni aklimatizace záleží na tom, zda implementoval Systém řízení rizik spojených s únavou (Fatigue Risk Management).



Obrázek č. 12: Podprogram pilot je v neznámém stavu aklimatizace (s i bez FRM)

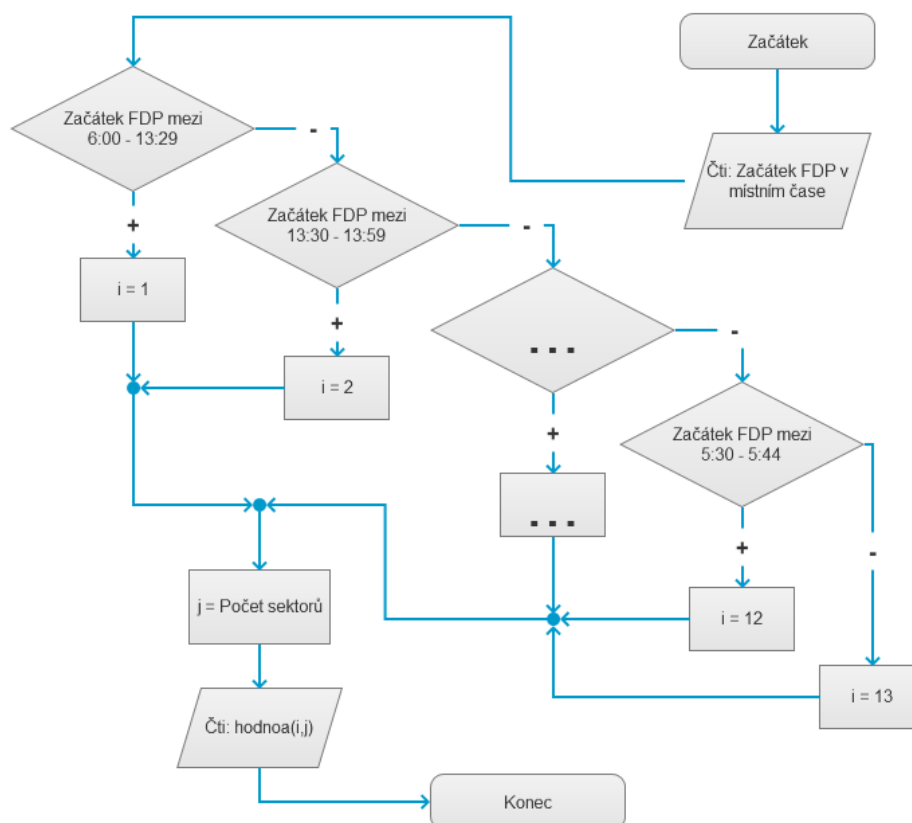
V obou případech již nezáleží na tom, kdy začíná doba letové služby. Celý výpočet je proto jednodušší a stačí znát jen počet plánovaných sektorů. Musí se však dbát na to, že se snižuje maximální počet plánovaných sektorů na 8. Pokud je počet sektorů vyšší, není letová služba proveditelná.

Jak již bylo napsáno, výpočet maximální doby letové služby pro pilota v neznámém stavu aklimatizace je stejný pro oba případy (pokud provozovatel implementoval FRM nebo ne). Liší se pouze v hodnotách maximální doby letové služby. Pokud má provozovatel metodiku pro sledování rizik spojených s únavou posádek, je logické, že tito piloti mohou mít v tomto stavu aklimatizace delší letovou službu, než kdyby tomu tak nebylo.

Výpočet maximální doby letové služby pro aklimatizované členy posádky se použije se podobá podprogramu matice aklimatizace. Opět se data z tabulky, tentokrát Tabulka 2, transformují na hodnoty matice FDP_{ij} , kde i taktéž reprezentují číslo řádku matice a nabývá hodnot $i = \{1, \dots, 13\}$ a j analogicky reprezentuje číslo sloupce matice, přičemž nabývá hodnot $j = \{1, \dots, 10\}$.

Po zadání začátku doby letové služby a přidáním odchylky od UTC se zjistí místní čas. Podle toho, v jakém časovém intervalu začátek FDP nastane, algoritmus se dozví hodnotu i . Po zadání počtu plánovaných sektorů je ihned známá hodnota j , neboť počet sektorů přímo koresponduje s počtem sloupců matice.

V tabulce 2 jsou sloupce $j=1$ a $j=2$ spojeny do jednoho sloupce, jelikož nabývají stejných hodnot. Aby se v algoritmu nemusela přidávat další podmínka, jsou sloupce 1 a 2 zvlášť, byť mají stejné hodnoty.



Obrázek č. 13: podprogram matice FDP

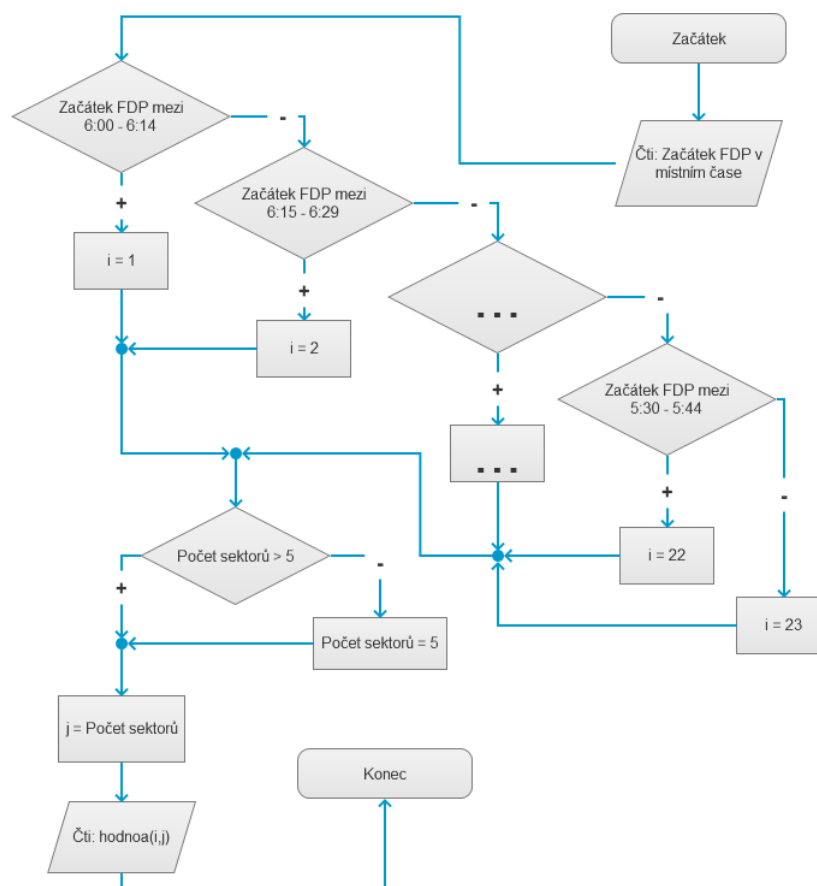
Nyní je známa základní doba letové služby bez jakéhokoliv prodloužení. Tato hodnota se bude později v algoritmu porovnávat se zadanou dobou letové služby, jestli nebyla maximální FDP překročena. Samozřejmě, že se tuto hodnotu lze dále modifikovat, a to zejména pomocí prodloužení letové služby.

3.3 Prodloužení letové služby

Prodloužit letovou službu lze hned dvěma způsoby. Tím prvním je možnost využití prodloužení bez odpočinku za letu. Výpočet je téměř totožný s výpočtem maximální doby letové služby. Opět se využije podprogramu. Tentokrát s názvem Matice FDP s prodloužením bez odpočinku za letu.

Stejně jako i v předchozích případech je definována matice $EFDP_{ij}$, kde i reprezentuje počet řádků matice a nabývá hodnot $i = \{1, \dots, 23\}$ a j reprezentuje počet sloupců matice a nabývá hodnot $j = \{1, 2, 3, 4, 5\}$. Hodnoty matice jsou definovány podle tabulky 5 a uloženy v databázi.

Dělení časů začátků FDP je daleko jemnější, což zapříčinilo, že se v této matici se oproti matici FDP_{ij} zvýšil počet řádků až o 10. Naopak se snížil počet možných sektorů a s tím i počet sloupců matice na 5. Opět, jako i v předchozím případě, v tabulce 5 jsou údaje pro 1 a 2 sektory spojeny v jednom sloupci. V matici algoritmu jsou však tyto hodnoty rozděleny dodvou sloupců pro každý sektor zvlášť.



Obrázek č. 14: podprogram FDP s prodloužením bez odpočinku za letu

Na základě začátku doby letové služby se určí hodnota i . Tento první krok je stejný jako u předchozího podprogramu. Uživatel mohl na začátku zadat větší počet plánovaných sektorů, než je, v tomto případě, limit 5. Z toho důvodu se jakákoliv vyšší hodnota automaticky změní na maximální možný počet sektorů, což je 5. Další postup je již známý. Na základě počtu sektorů se určí i pořadí sloupce j . Výsledná hodnota matice $EFDP_{ij}$ je novým maximálním FDP.

Prodloužení doby letové služby s odpočinkem za letu nelze kombinovat s prodloužením FDP bez odpočinku za letu. O tuto podmínku se v algoritmu postará rozhodovací blok „*Prodloužení FDP bez odpočinku za letu*,“ kde při kladném rozhodnutí „+“ proběhne výše popsáný podprogram. Při záporném rozhodnutí „-“ algoritmus přejde k dalšímu rozhodovacímu bloku „*Prodloužení FDP s odpočinkem za letu*.“ Tím je zajištěno, že proběhne výpočet maximální doby letové služby pouze pro prodloužení bez odpočinku za letu nebo pro prodloužení s odpočinkem za letu. Popřípadě je zde i třetí možnost (záporné rozhodnutí u posledně zmíněného rozhodovacího bloku), kdy se neuvažuje s žádným prodloužením.

Aby bylo možno využít prodloužení s odpočinkem za letu, musí se nejdříve zkontrolovat, jestli byl odpočinek, který plánované letové službě předchází, trval alespoň 14 hodin. Pokud je vše v pořádku, tak se pokračuje dál. Předchozí letová služba mohla ale mohla trvat déle než jen 14 hodin, a proto je třeba brát v potaz i tuhle situaci. Je-li vše v pořádku, může se přistoupit k výpočtu maximálního FDP. Pokud se stane, že jedna z podmínek není splněna, nelze tuto službu prodloužit s využitím odpočinku za letu. Zobrazí se zpráva: „Nelze prodloužit z důvodu nedostatečného předešlého odpočinku“ a vrátí se zpět na začátek algoritmu, aby se zadala nová vstupní data.

Nyní je potřeba zvolit, kolik pilotů zesílí posádku. Letící piloti by přeci nemohli odpočívat, kdyby je nikdo nenahradil. Proto je prodloužení maximální doby letové služby s odpočinkem za letu spjato se zesílenou posádkou.

V případě jednoho pilota navíc se dále určuje třída zařízení odpočinku za letu a začíná se od „nejméně pohodlné“ třídy 3. V případě zvolení této třídy se zvyšují maximální FDP na 14 hodin. V opačném případě je zde rozhodovací blok pro 2. třídu. Pro kladnou odpověď je limit doby letové služby zvýšen ještě o jednu hodinu na 15 hodin. Nevybere-li se třída 3 ani 2, algoritmus automaticky předpokládá, že je vybrána poslední zbývajících třída 1, limit FDP se navýší na 16 hodin a pokračuje se dál.

Posádku lze také zesílit dvěma piloty navíc. Algoritmus také myslí na to, co by se stalo, kdyby byla posádka zesílena o více než dva piloty. Taková situace by neměla na výpočet maximální doby letové služby žádný vliv. Z toho důvodu je (pouze pro potřeby výpočtu) maximální počet pilotů, kteří mohou zesílit posádku snižen na dva piloty.

Další výpočet probíhá analogicky jako v případě zesílení posádky jedním pilotem. Opět se musí rozhodnout, která třída zařízení určené k odpočinku za letu je vybrána. Obecně platí, že všechny limity pro dobu letové služby se ještě navýší o jednu hodinu oproti předchozím situacím.

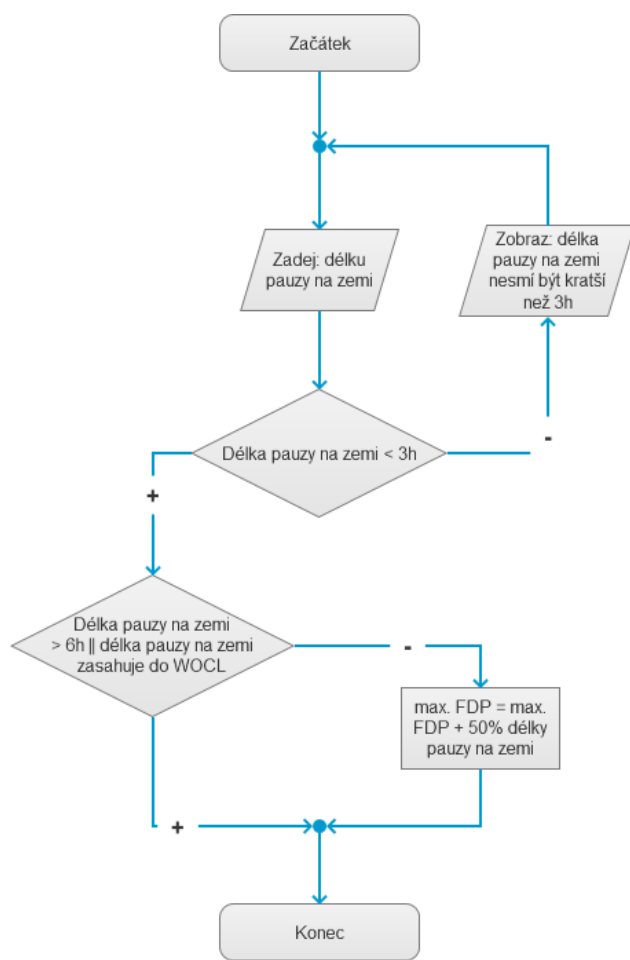
3.4 Zkrácení letové služby

Obecně platí, že se bude provozovatel snažit maximální dobu letové služby spíše prodloužit. Existuje však situace, kdy se musí přistoupit k přesně opačnému kroku. K takovému případu nastává, když byl zkrácen odpočinek, jenž předcházel plánované službě.

Nezbývá tedy nic jiného, než zkrátit letovou službu právě o tolik o kolik byl předcházející odpočinek zkrácen. Předcházel-li řádný odpočinek, algoritmus pokračuje beze změny dál. Samozřejmě, že se musí také prodloužit následující odpočinek, to je ale úkol pro pozdější bloky algoritmu.

3.5 Dělená služba

Existuje ještě jedna možnost, jak nepřímou prodloužit maximální dobu letové služby. Proveďte se to pomocí dělené služby. Musí se ale nejdříve splnit podmínka, že se tato možnost nebude kombinovat s prodloužením letové služby. Jsou-li všechny podmínky splněny, přechází algoritmus k podprogramu *Dělená služba*.



Obrázek č. 15: podprogram Dělená služba

S dělenou službou je spjata přestávka na zemi, která se zpravidla započítává do FDP. Proto se algoritmus ze všeho nejdřív zeptá uživatele, jak dlouhá jeho přestávka bude. Platí, že přestávka na zemi musí trvat alespoň 3 hodiny. Je-li přestávka kratší, zobrazí se výstražná zpráva informující o nedostatečné délce pauzy a algoritmus se vrací zpět ke kroku, kde se zadává plánovaná přestávka na zemi.

V dalším kroku se rozhodne, jestli se bude přestávka započítávat do doby letové služby. Pokud se splní alespoň jedna z následujících podmínek, čas strávený přestávkou není do FDP započítán. Tato podmínka je buďto taková, že je trvání přestávky delší než 6 hodin nebo že zasahuje okna

cirkadiánního útlumu. Pokud naopak ani jedna z podmínek neplatí, do doby letové služby se započítává 50 % z přestávky.

Důvod, proč je výše napsáno, že se doba letové služby nepřímou prodlouží je proto, že s maximálním FDP se nijak nehýbe. Protože má ale pilot možnost přestávky, je v práci fakticky déle, než by za normálních okolností byl.

3.6 Kumulativní doba letové služby

Nyní již algoritmus prošel všechny možnosti, jak spočítat maximální možnou délku letové služby. Zbývá už jen porovnat plánovanou FDP a maximální FDP. Samozřejmě, že plánovaná FDP nesmí přesáhnou maximální možnou dobu letové služby. Pokud by taková situace nastala, tak by se zobrazila zpráva, že byla naplánovaná taková letová služba, jež je delší než povolená FDP. Zároveň by algoritmus uživatele vyzval, aby zkrátil plánovanou FDP a vrátil se zpět na začátek, kde se zadávají vstupní údaje.

I v situaci že by bylo všechno správně, jsou zde ještě jisté omezující faktory a těmi jsou kumulativní doby letové služby. Ty se počítají pro tři různě dlouhá období a jejichž hodnoty byly pro účely tohoto algoritmu pojmenovány jako C1, C2 a C3.

Výpočet je proveden tak, že se sečtou doby letových služeb za posledních 28 dní pro C1, za kalendářní rok pro C2, a nakonec součet za posledních 12 měsíců po sobě jdoucích pro C3. Všechny výpočty jsou včetně právě plánované letové služby. Ostatní data jsou získána z databáze dob letových služeb.

Aby mohl algoritmus úspěšně pokračovat s dalšími operacemi, musí být splněny všechny podmínky. Těmito podmínkami jsou:

- $C1 \leq 100 \text{ h}$
- $C2 \leq 900 \text{ h}$
- $C3 \leq 1000 \text{ h}$

Stane-li se, že bude platit, byť jen jedna z těchto nerovností, zobrazí se zpráva, že bylo překročeno maximální kumulativní FDP. Zároveň algoritmus také upozorní na to, která z hodnot byla překročena. Protože se nesmí v takovém případě plánovaná letová služba provést, vrátí se algoritmus zpět na krok, kde se zadávají vstupní data.

3.7 Výpočet doby služby

Co se týče výpočtu maximální doby služby, předpis v tomto směru neurčuje moc omezujících podmínek. Víceméně je na provozovateli, jaká pravidla si nastaví v rámci svého plánování.

Již na úplném začátku algoritmu si uživatel vybírá, jaký typ služby má naplánovaný. Pokud si zvolí standby službu, algoritmus potřebuje vědět, kde tato služba bude probíhat. Jestliže se totiž jedná o letištní standby, je zde limitující podmínka, že součet doby strávené ve standby a případné doby letové služby je povolený na maximálně 16 hodin a započítává se plně jako doba služby. Naopak kdyby byla standby služba prováděna mimo letiště, takováto služba také nesmí být delší jak 16 hodin, pro výpočet doby služby se ale započítává pouze 25 % z této služby.

Z definice doby služby vyplývá, že do sebe zahrnuje i letovou službu. Musí proto být pokaždé delší nebo alespoň stejně dlouhá jako letová služba. Kdyby se stalo, že by pilot

třeba omylem zadal dobu služby kratší než dobu letové služby, algoritmus automaticky zvýší DP na hodnotu FDP.

Ještě, než se přejde k výpočtům kumulativních dob služeb, zkontroluje se, jestli platí tvrzení uvedené v předchozím odstavci. Během standby služby musí být její trvání kratší nebo stejné jako 16 hodin. Pokud tato podmínka není splněna, respektive je splněna podle zakreslení v algoritmu, zobrazí se zpráva. Ta obsahuje informaci o tom, že byla překročena tento 16 hodinový limit, přičemž navrhne opravná opatření. Těmi jsou buďto zkrácení doby služby nebo doby letové služby a vrátí uživatele zpět na začátek algoritmu k zadání nových údajů.

3.8 Kumulativní doba služby

Stejně jako u letové služby se musí zkontrolovat, jestli se nenakumulovala služba do příliš krátkého časového intervalu. Aby se této situaci předešlo, vytvořila se proto část algoritmu, která tyto rizikové služby rozpozná. Algoritmus si bude součty kumulativních dob služeb pamatovat pod názvy CDP1, CDP2 a CDP3.

Kumulativnost dob služeb se kontroluje v poměrně kratších intervalech, než je tomu u FDP. První hodnota CDP1 provádí součet předešlých služeb včetně té plánované pro posledních 7 dní po sobě jdoucích. CDP2 dále počítá posledních 14 dní a CDP3 posledních 28 dní po sobě jdoucích.

Aby bylo algoritmu umožněno pokračovat v dalších výpočtech, musí být splněny všechny tyto podmínky:

- $CDP1 \leq 60 \text{ h}$
- $CDP2 \leq 110 \text{ h}$
- $CDP3 \leq 190 \text{ h}$

Stačí, aby jediná z těchto podmínek nebyla splněna a zobrazí se zpráva „překročeno maximální kumulativní DP...“ Dále algoritmus navrhne možnou změnu, a to ve zkrácení doby služby, přičemž také ukáže, která hodnota CDP byla překročena. Uživatel se vzápětí vrátí zpět na začátek algoritmu, aby mohl zadat nová správná data.

3.9 Výpočet doby odpočinku

Pokud jsou známy limity pro dobu služby a dobu letové služby, může se přejít k finální části algoritmu a určit tak potřebnou dobu odpočinku. Výpočet základní doby odpočinku je přímo spjat s předchozí letovou službou a také s místem, kde bude aplikován.

Prvním rozhodovacím blokem v této části algoritmu je sice rozhodnutí o tom, bude-li se jednat o zkrácený odpočinek nebo ne. Ve většině případu půjde nejspíš o řádný odpočinek, proto bude důležitější rozhodovací blok „*home base*“ nebo domácí základna či mateřské letiště.

Algoritmus počítá dobu odpočinku, která musí nastat po jakékoliv naplánované službě. Další služba proto může nastat, až po minimální době odpočinku, která se určí tímto algoritmem. Je-li tedy odpočinek prováděn v home base a skutečná služba trvala déle jak 12 hodin, odpočinek musí trvat alespoň tak dlouho jako předcházející služba. V ostatních případech, kdy je FDP kratší než 12 hodin, je minimální doba odpočinku nejméně dvanáctihodinová.

Zcela analogicky se postupuje v případě, kdy je odpočinek prováděn mimo home base. V rozhodovacím bloku se tentokrát určuje, zdali skutečná DP nebyla delší než 10 hodin. Při záporném rozhodnutí je minimální doba odpočinku 10 hodin a při kladném stejně jako v předchozím případě trvá odpočinek alespoň tak dlouho, jako služba, kterému předcházela.

Jsou případy, kdy je potřeba z různých důvodů odpočinek zkrátit. Je to možné, ale za určitých podmínek. Předně nesmí zkrácený odpočinek nastat hned po jiném zkráceném odpočinku. Kdyby se takový odpočinek naplánoval, zobrazí se zpráva „Nelze zkrátit, protože byl předchozí odpočinek již zkrácen“ a algoritmus se vrátí zpět před rozhodovací blok „*Zkrácený odpočinek*.“

Jestli nebyl předchozí odpočinek zkrácen, může se tento plánovaný zkrátit, avšak jenom za podmínky implementovaného Fatigue Risk Managementu. Jinak krátit nelze, což se také objeví v informační zprávě a algoritmus se vrací před ten samý rozhodovací blok jako v předchozím případě.

Nakonec, jsou-li obě podmínky splněny, odpočinek se může zkrátit. V předpise ale není uvedeno minimální doba odpočinku při zkrácení. Tu si totiž určuje provozovatel

a algoritmus tuto hodnotu tedy nemusí znát. Buďto bude na zodpovědnost pilota si tuto mez uhlídat, nebo si bude muset tuto hodnotu definovat a uložit do databáze.

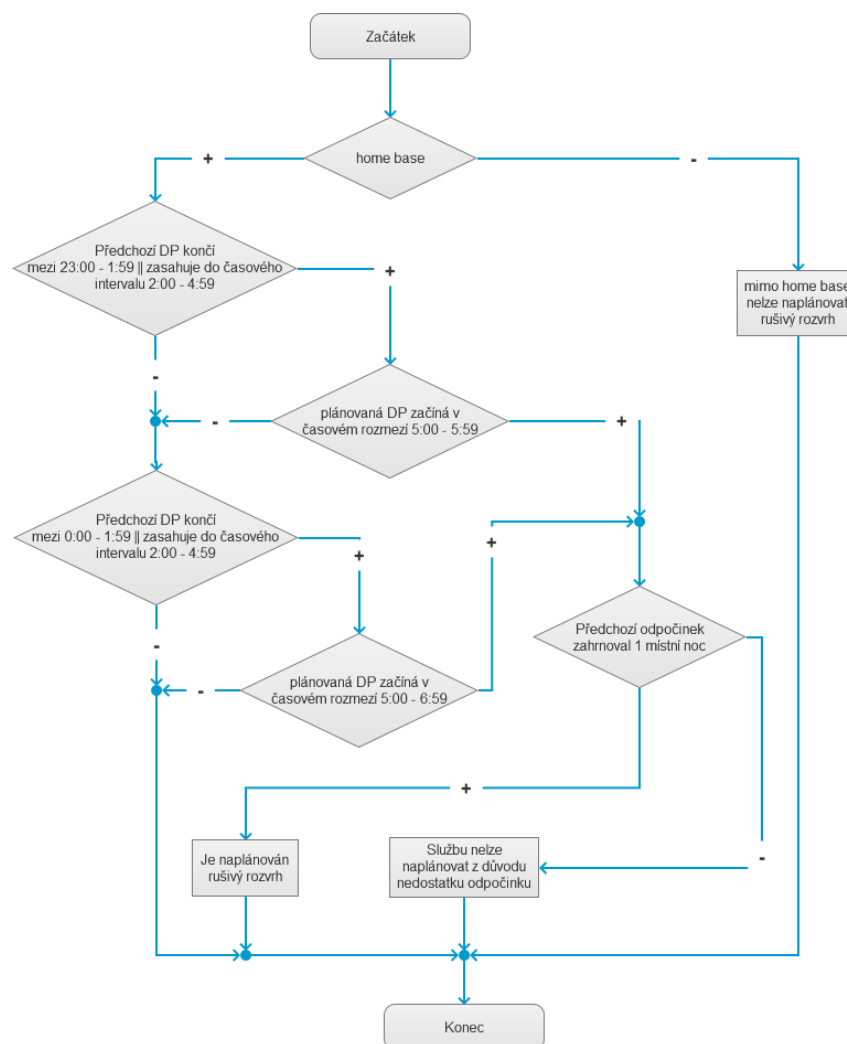
3.10 Rušivý rozvrh

Plánovat „rušivý rozvrh“ se může jenom v home base. Mimo toto mateřské letiště proto odpadá možnost dalšího plánování s využitím rušivého rozvrhu. V home base je situace úplně jiná. Algoritmus nejdříve prozkoumá časný typ rušivého rozvrhu. Pokud služba končí v časovém intervalu mezi 23:00 a 1:59 dalšího dne nebo služba nějak zasahuje do noční služby (období mezi 2:00 a 4:59), pokračuje algoritmus dále.

Je jasné, že není možné naplánovat další službu několik minut hned po skončení jedné. Má-li navíc další služba začínat brzy, to jest v čase mezi 5:00 a 5:59 ráno, pilot si potřebuje dostatečně odpočat. Aby mohla být plánovaná služba proveditelná, je zapotřebí, aby předchozí odpočinek obsahoval alespoň jednu místní noc.

Krom časného typu je potřeba také zkontrolovat pozdní typ rušivého rozvrhu. Ten nastává, pokud předchozí služba končí v rozmezí půlnoci a 1:59 hodin ranních nebo zasahuje do noční služby a zároveň plánovaná služba začíná v intervalu mezi 5:00 a 6:59 hodin. I v tomto případě musí odpočinek mezi minulou a plánovanou službou obsahovat minimálně jednu místní noc.

Je-li podmínka dostatečného odpočinku splněna, algoritmus ví, že se je naplánován rušivý rozvrh. Tato informace může být důležitá pro další výpočet doby odpočinku. V druhém případě, kdy nebyl odpočinek dostatečný algoritmus vyhodnotí, že plánovanou službu nelze provést z důvodu nedostatečného odpočinku. Tím je podprogram ukončen a může se přejít na další podmínky algoritmu.



Obrázek č. 16: podprogram rušivý rozvrh

Jako první úkon, jenž algoritmus provede hned po ukončení podprogramu je takový, že vyvolá informace z předcházejícího podprogramu a zeptá se na dostatečný odpočinek. Při záporné odpovědi se zobrazí zpráva o tom, že nelze na požadovaný čas naplánovat službu z důvodu nedostatečného odpočinku a přejde se na první krok algoritmu. Pokud je vše v pořádku, nic nebrání tomu přejít na další kroky.

3.11 Prodloužený odpočinek

Předpis stanovuje, že musí nastat prodloužený odpočinek (nejméně v délce 36 hodin) alespoň jednou během 7 dní, respektive jednou za 168 hodin. V algoritmu je ale podmínka definována trochu jinak. Je tomu tak z důvodu fungování algoritmu. Ten totiž sleduje při zadávání začátků DP a FDP, jestli byl dodržen dostatečný odpočinek mezi službami. Co je ale teď, tak zároveň plánuje minimální dobu odpočinku pro další službu.

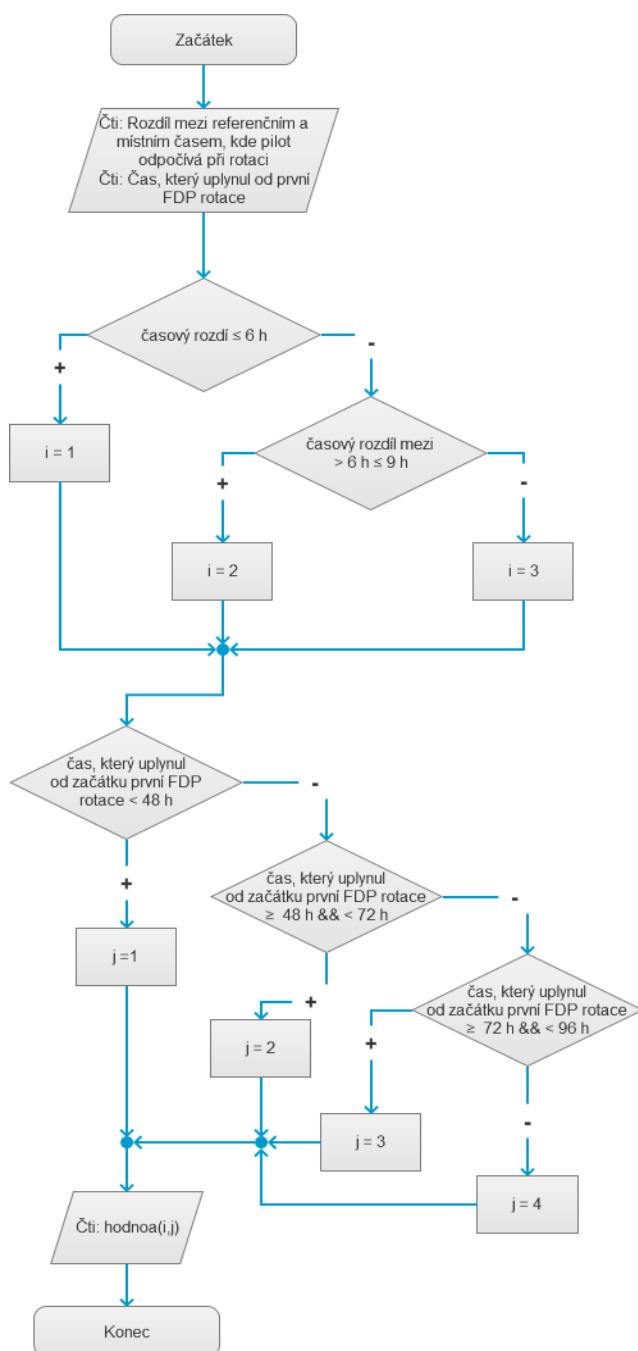
Pro lepší představivost může posloužit následující příklad. Od konce posledního prodlouženého odpočinku do konce zadané služby uběhlo 163 hodin. Než došel algoritmus k této podmínce, tak spočítal, že minimální doba odpočinku je například 12 hodin. Když se těchto 12 hodin přičte k 163 hodinám, výsledné číslo je větší, než 168 hodin. Proto se doba minimálního odpočinku změní na 36 hodin.

Dalším věc, kterou je potřeba brát v potaz, je podmínka, že alespoň dva prodloužené odpočinky musí zahrnovat dva místní dny. Tato podmínka byla rozdělena do dvou rozhodovacích bloků. První dává při plánování takového odpočinku ještě určitou časovou rezervu. Pokud ale tento odpočinek nastane až v posledním možném termínu, další rozhodovací blok již zajistí, že další prodloužený odpočinek bude také zahrnovat dva místní dny. Pokud je vše bez v pořádku, může se pokračovat dále.

V následujícím rozhodovacím bloku se využije informací z podprogramu Rušivý rozvrh. V případě, že by byla naplánovaná služba podle rušivého rozvrhu 4x během období mezi dvěma prodlouženými odpočinky, prodloužený odpočinek se ještě více prodlouží. Pro pilota musí být takový rozvrh služeb nesmírně namáhavý, proto je minimální doba odpočinku alespoň 60 hodin.

3.12 Rotace

Při rotaci se rozhoduje o počtu místních nocí, které člen posádky musí strávit v home base. Opět je problém řešen pomocí podprogramu. Podprogram Rotace je řešen pomocí matice $Rotace_{ij}$, a tudíž se i podobá některým dřívějším podprogramům. Pořadí řádku opět reprezentuje hodnota i , jež zde nabývá pouze hodnot $i = \{1, 2, 3\}$. Hodnota j analogicky k předchozím případům nabývá hodnot $i = \{1, 2, 3, 4\}$, a reprezentuje pořadí sloupce.



Obrázek č. 17: podprogram Rotace

K výpočtu je třeba znát dvě hodnoty. To první je rozdíl mezi referenčním časem a místním časem, kde pilot během rotace odpočívá. Podle velikosti tohoto časového rozdílu se rozhodne o hodnotě i , a tudíž i o řádku matice.

Druhá hodnota reprezentuje čas od začátku doby letové služby, kterou zároveň započala rotace. Na základě takto uplynulé doby algoritmus určí, který sloupec matice je platný. To zjistí díky hodnotě j .

Matice $Rotace_{ij}$ je jako všechny ostatní matice definována v databázi. Její hodnoty odpovídají Tabulce 7. Výsledná hodnota i a j odpovídá hodnotě v řádku a sloupci matice.

3.13 Konec algoritmu

Ještě předtím, než algoritmus došel k problematice rotace, prošel přes rozhodovací blok „*Předchozí odpočinek byl zkrácený.*“. Tahle podmínka je důležitá proto, aby se nezapomnělo přičíst zkrácený čas z předchozího odpočinku k tomu nastávajícímu. A přesně tato operace se provede při záporné odpovědi rozhodovacího bloku.

Poté co ještě algoritmus projede přes rotace, dostane se k poslednímu rozhodovacího bloku. Zde se konečně zkontroluje, jestli byla služba naplánovaná tak, že se dodržel minimální odpočinek po předchozí službě. V případě, že se nebyl odpočinek dostatečný, zobrazí algoritmus zprávu a tomto zjištění a vyzve k naplánování doby služby a doby letové služby na jiný čas. Tím pádem se z úplného konce vrátí algoritmus zpět na úplný začátek.

Samozřejmě je zde i druhá možnost, a to taková, že je vše v pořádku. V tom případě se všechny relevantní vypočtené hodnoty uloží do databáze a budou sloužit k budoucím výpočtům.

4 Ukázka vytvořeného řešení

Program je vytvořen v softwaru Microsoft Excel. Oproti běžným souborům Excelu se liší v tom, že se musí při spouštění povolit makra [10]. Program je přílohou diplomové práce. Obsahuje 7 listů. Šest jich je určených pro ukázkou dat pilotů z modelové situace (viz níže) a jeden list je prázdný pro nové zápisy. Soubor svým naprogramování reprezentuje algoritmus pro sledování zatížení letových posádek, zároveň je také databází, kam se ukládají všechny údaje.

4.1 Představení programu

Všechny potřebné údaje pro jednoho pilota se vlezou na jeden list souboru. Pro každou jednotlivou informaci je určena jedna buňka. Informace jsou řazeny po sloupcích pro větší přehlednost.

Vrchní část listu tvoří horní lišta. Úplně nahoře lišty je modré tlačítko NEW RECORD, pomocí kterého se provádí každý nový zápis. Jedná se také o jediný aktivní prvek celého listu. Zelená lišta pod tlačítkem NEW RECORD je jedinou buňkou, kterou lze přepisovat. Uživatel si tam napíše své jméno.

NEW RECORD																
NAME SURNAME																
#	DATE	FLIGHT DUTY PERIOD				No. OF SECTORS	TYPE OF DUTY	DUTY PERIOD					REST PERIOD min. DUR.	DURATION OF		REMARK
		UTC		LOCAL				UTC		LOCAL				FDP	DP	
		START	END	START	END			START	END	START	END					
1																
2																
3																
4																
5																

Obrázek č. 18: vizuální podoba programu

Další buňky v horní liště už jen slouží k pojmenování daného sloupce listu. Jako první zleva je #. To představuje číslo přiřazené ke každému zápisu. Dále vpravo je sloupec s názvem DATE. Již podle názvu je jasné, že se zde bude zobrazovat datum plánovaných služeb.

Další část lišty je určena době letové služby. Uživatel může vidět kdy začala a skončila jeho letová služba, a to jak v čase UTC, tak i v místním čase. Nesmí také chybět důležitá informace o počtu uletěných sektorů.

Prostřední část lišty se podobá předchozí části FDP. Zde se řeší doba služby. Opět zde může uživatel vidět začátek a konec své služby v obou časech (UTC i místní). Místo počtu sektorů je zde ale sloupec s názvem druh služby (TYPE OF DUTY).

Následuje prázdné žluté políčku označuje sloupec, kde se zobrazují dodatečné informace pro pilota o jeho službě. Mezi tyto informace může patřit například místo vykonání odpočinku, informace o prodloužení letové služby nebo hlášení upozorňující na chybu.

Dalším sloupcem je doba odpočinku, respektive minimální doba odpočinku, která se musí dodržet od současné služby po naplánování další. Taktéž se vpravo od doby odpočinku zobrazí trvání doby letové služby a doby služby, které byly naplánovány. Jedná se pouze o rozdíl konce a začátku FDP a DP.

Poslední sloupec je určen pro poznámky. Zde si uživatel může zapsat například letiště, na kterých během své letové služby přistál. Popřípadě tam může zapsat i jiné textové informace, je však omezen šířkou buňky.

Největší část listu ale tvoří samotné pole pod spodní lištou. Je zde místo až na 350 zápisů. Při dalším novém zápisu se ten poslední automaticky smaže. Jak již bylo řečeno, pole slouží pouze vizuální prohlídce dat. Data jsou totiž zapsány v určitém formátu a o určité hodnotě tak, aby sloužily budoucím výpočtům. Za žádnou cenu proto nesmí dojít k jejich přepsání.

4.2 Uživatelské rozhraní

Nyní již k vstupním datům. Kliknutí na tlačítko NEW RECORD vyvolá nové okno, do které lze tato data zapsat. Jako první informaci zadá uživatel stupeň svojí aklimatizace. Na výběr je ze tří možností, které jsou již známy z popisu přepisu normy letového zatížení posádek.

Všechny časové údaje se musí zapsat v přesném formátu, jinak nebude program fungovat správně. Datum se zapisuje ve formátu *dd-mm-yyyy*. Pro některé dny a většinu měsíců se může datum zapsat jako *d-m-yyyy*, důležitá je pomlčka, která den, měsíc a rok odděluje.

Hodnota času naopak podporuje formát *hh:mm*. Pro některé časy je také možnost *h:mm*. Důležité je dodržovat pomlčku. Časové hodnoty v tomto formátu se zapisují do políček Start of FDP/DP a End of FDP/DP.

Initial Data

Acclimatization

Date (dd-mm-yyyy)

Start of FDP End of FDP

Start of DP End of DP

UTC +/- Duty type

No. of Sectors

Remarks

☐ Rest away from Home Base

☐ Reduced Rest

☐ Extended FDP without in-flight rest

☐ Extended FDP with in-flight rest

Apply **Correct** **Acclimatisation**

Obrázek č. 19: okno k zadání vstupních dat

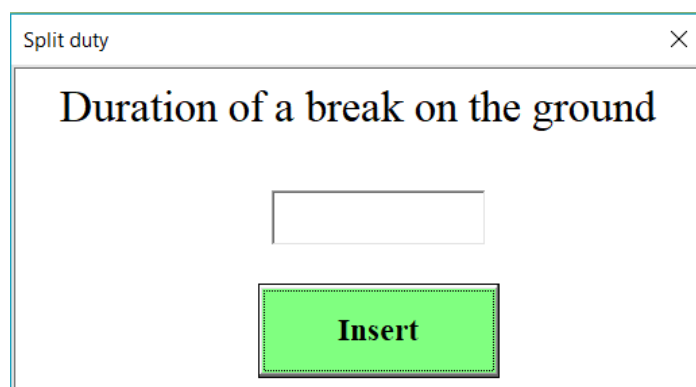
Odchylka od UTC nezapisuje. Namísto toho se přidávají tlačítka. Každý klik tlačítka zvýší nebo sníží hodnotu o jedno celé číslo. Jako předvolená hodnota je nula. Maximální kladná i negativní odchylka je do ± 12 . Dalším kliknutím se již číslo nezvýší, respektive nesníží.

Políčko *No. of Sectors* (česky počet sektorů) funguje na stejném principu jako Odchylka UTC. Přírůstky jsou vždy celá čísla. Liší se ale v tom, že předvolená hodnota je nejmenší možný počet sektorů, tudíž 1. Maximální hodnota se však liší v závislosti na tom, co je dále zvoleno. Ve většině případů je nejvyšším číslem 10. V případě prodloužení doby letové služby bez odpočinku za letu klesá maximální počet sektorů na 5 a po vybrání prodloužení FDP s odpočinkem za letu tato hodnota klesá na 3 sektory.

Výběr typu služby se provede u políčka *Duty type*. Zde se ze seznamu vybere z možností:

- FDP + pre/post flight duty = FDP + předletová a poletová služba
- Standby = letová záloha
- Positioning = přemístování
- Split Duty = dělená služba
- Other = jiná služba

Pokud se vybere dělená služba, zobrazí se nové okno, kde se musí zapsat délka pauzy na zemi. Je to povinný údaj, jelikož se podle délky této přestávky se pak mění i délka doby letové služby. Formát je stejný jako u předešlých časových údajů (hh:mm).



Obrázek č. 20: okno Split Duty

Poslední největší políčko je *Remarks*. Nemá vyhrazený žádný konkrétní formát. Je určeno převážně pro text. Jediné omezení je v počtu možných znaků, které lze zapsat.

Krom polí, do kterých se zapisují hodnoty jsou v pravém spodním rohu zatrhávací políčka. Každé vybrané políčko nějak ovlivňuje výpočet FDP, DP a doby odpočinku. Zatrhuje se některá z těchto možností:

- Rest away from home base = odpočinek mimo home base
- Reduced rest = zkrácený odpočinek
- Extended FDP without in-flight rest = prodloužená FDP bez odpočinku za letu
- Extended FDP with in-flight rest = prodloužená FDP s odpočinkem za letu

U posledních třech možností platí, že může být zatrhnuta pouze jedna. Nelze prodloužit dobu letové služby bez odpočinku za letu zároveň i s využitím odpočinku za letu. Dále také nemůže být následující odpočinek zkrácený, pokud mu předcházela prodloužená letová služba.

Po zatrnutí „*Extended FDP with in-flight rest*“ není ještě všemu konec. Zobrazí se nové okno, kde si uživatel vybere zesílení posádky a třídu zařízení pro odpočinek. Uživatel si samozřejmě nemůže vybrat dva typy zařízení najednou nebo oba druhy zesílení posádky, vždy jenom jednu možnost. Protože zesílení posádky o více než 2 členy nemá na výpočet maximální FDP žádný vliv, jsou v okně pouze 2 možnosti.

Extended FDP with in-flight rest

☐ 1 additional flight crew member ☐ Class 1 rest facility

☐ 2 additional flight crew members ☐ Class 2 rest facility

☐ Class 3 rest facility

OK

Obrázek č. 21: okno prodloužení FDP s odpočinkem za letu

Pokud je vše vybráno a zadáno ve správném formátu, zbývá už jen kliknout na tlačítko *Apply*. V tom okamžiku se program provede všechny výpočty tak, jak jsou zapsány v algoritmu. Všechny zápisy v databázi se posunou o jeden řádek níž a na místo #1 se dostane nový zápis. V případě překročení nějakého limitu se zobrazí varovná zpráva (viz kapitola 4.4 Ukázka fungování algoritmu).

V okně vstupní data se dále ještě objevují dvě tlačítka. Po kliknutí na *Correct* se objeví nové okno. Zde uživatel může upravit předchozí zápisy, pokud si všimne nějaké chyby. Okno pro snadnou orientaci mezi zápisy připomíná hlavní stránku souboru.

Vyhledávání požadovaného zápisu probíhá dvěma způsoby. Pokud se jedná o pozdější zápis, je výhodné zapsat číslo zápisu # a kliknout na *FIND #*. Druhá možnost měnit zápisy pomocí tlačítek *PREVIOUS* (předchozí) a *NEXT* (další). Tato tlačítka posunou zápis o 1 nahoru či dolů.

Uživatel tedy změní jaké hodnoty chce, samozřejmě při tom nesmí měnit formát zápisu. Poté klikne na *SAVE*. Provede se opět kontrola překročení povolených limitů podle algoritmu pro sledování dodržování normy letového zatížení posádek. Tlačítko *DELETE LAST RECORD* slouží pro smazání posledního zápisu, který je například nějak vadný (myšleno, že nesplňuje nějakou podmínku pro bezchybný provoz služby) a uživatele si nepřeje tento zápis nijak dále upravovat.

Correction

#	Date	Flight Duty Period				No. of Sectors	Type of Duty	Duty Period					Rest Period	Duration of			Remark
		UTC		Local				UTC		Local				min. time	FDP	DP	
		Start	End	Start	End			Start	End	Start	End						
1	29-11-2017	09:40	18:30	10:40	19:30	2	FDP + pre/post flight Duty	09:20	18:45	10:20	19:45	HB	12:00	08:50	09:25		LKMT - LFST - LKMT
2	25-11-2017	05:40	17:40	06:40	18:40	3	FDP + pre/post flight Duty	05:35	18:00	06:35	19:00	HB	12:25	12:00	12:25		LKMT - ELLX - EBAW - LKMT
3	24-11-2017	07:30	15:30	08:30	16:30	3	FDP + pre/post flight Duty	07:20	15:55	08:20	16:55	HB	12:00	08:00	08:35		LKMT - EDDC - EPSC - LKMT

FIND #

PREVIOUS

NEXT

DELETE LAST RECORD

SAVE

Obrázek č. 22: okno Correction

Poslední tlačítko v okně Vstupní data *Acclimatization* slouží k tomu, když si uživatel není jistý stupněm aklimatizace. Zadávají se zde dva časy. Jsou jimi konec poslední služby a začátek další služby, každá služba s jejích příslušným datumem. Časy jsou UTC, takže je nutné přidat i odchylku od UTC. Jako čas konce poslední služby i s příslušným datem a odchylkou UTC se automaticky zobrazí hodnoty z posledního zápisu. Stačí už jen dopsat hodnoty pro další plánovanou službu. Přesto se dají i hodnoty konce poslední služby přepsat.

Po kliknutí na CALCULATE se provede výpočet, jaký je popsán v podprogramu „matice Aklimatizace.“ Uživatel teď zná stupeň aklimatizace. Tato informace se navíc automaticky přesune do okna „Vstupní data“.

Defining State of Acclimatisation

Insert Data

End of the last duty *Start of the next duty*

Date (dd-mm-yyyy) Date (dd-mm-yyyy)

End of DP (UTC) Start of DP (UTC)

UTC +/- UTC +/-

CALCULATE

Obrázek č 23: okno Určení stupně aklimatizace

4.3 Modelová situace

Fungování algoritmu lze nejlépe ukázat na konkrétní situaci. K tomuto účelu byla vytvořena fiktivní letecká společnost. Jedná se o malou firmu, která disponuje dvěma letouny L_1 a L_2 . S těmito stroji létá dohromady 6 pilotů ($P_1, P_2 \dots P_6$). Každý z nich má také typová výcvik na různé letouny.

- Pilot 1 - L_1
- Pilot 2 - L_2
- Pilot 3 - L_2
- Pilot 4 - L_1, L_2
- Pilot 5 - L_1, L_2
- Pilot 6 - L_1, L_2

Na základě svých typových kvalifikací potom vzniknou 3 posádky. Piloti P_1 a P_4 obsluhují letoun L_1 . Piloti P_2 a P_3 obsluhují letoun L_2 a nakonec poslední posádka s piloty P_5 a P_6 je schopna létat s oběma stroji.

Všechny předchozí lety posádek jsou zasány v programu v příloze B. Úkolem je naplánovat letovou službu pro jednu z posádek s letounem L_1 , jež má následujícími parametry:

- Datum: 1.12.2017
- Trasa: LKMT-LKTB-LSGG
- Počet sektorů: 3
- ETD (LKMT): 8h 30min
- ETA (LSGG): 15h 30min
- DP: 8h 15m – 15h 50min

Druhý let s letounem L_2 má následující parametry:

- Datum: 1.12.2017
- Trasa: LKMT-EVRA-ESMS-EDDC-LKMT
- Počet sektorů: 4
- ETD (LKMT): 6h 30min
- ETA (LKMT): 19h 20min
- DP: 6h 10min – 19h 35min

4.4 Ukázka fungování programu

První letovou službu mohou splnit pouze první (P₁ a P₄) a třetí posádky (P₅ a P₆). Jen tyhle posádky jsou totiž schopny obsluhovat letoun L₁. Při zadání vstupních hodnot pro členy první posádky a kliknutí na tlačítko „Apply“ se zobrazí zpráva, že kumulativní letová služba za poslední kalendářní roky byla překročena o 1 hodinu.



Cumulative Flight Duty Period must be less or equal to 900:00 in a calendar year. The limit is exceeded by the value of: 1:00

To znamená, že tato posádka může vykonat v tomto kalendářním roce už jen jednu službu, která nebude delší než 6 hodin. Nezbyvá tedy než vymazat poslední vadný zápis tlačítkem DELETE LAST RECORD.

Třetí posádce hrozí, že 1.2.2017 v čase 8:30 UTC nebudou moci odstartovat, neboť po skončení služby 29.11.2017 v 18:45 UTC následuje prodloužený odpočinek alespoň 36 hodinový. Po zadání vstupních dat se však žádná výstražná zpráva nezobrazuje, takže je vše v pořádku. Zápis se ale ještě vymaže, neboť se bude ještě zadávat jiný.

Druhou letovou službu mohou splnit pouze druhá (P₂ a P₃) a třetí (P₅ a P₆) posádka. Po zadání vstupních hodnot pro členy druhé posádky nastává hned několik problémů. Posádka překročila kumulativní limit doby služby 100 hodin za posledních 28 dní.



Cumulative Flight Duty Period must be less or equal to 100:00 in 28 consecutive days. The limit is exceeded by the value of: 7:55

Po náhledu do databáze je snadné zjistit, že další služba je možná až 7.12.2017. Kdyby tohle nebyla pravda, tak či tak by se musel let odložit, protože je posádka překročila i limit kumulativní doby služby 60 hodin za posledních 7 dní o 2 a čtvrt hodiny.



Cumulative Duty Period must be less or equal to 60:00 in 7 consecutive days. The limit is exceeded by the value of: 2:15

V případě pouze tohoto omezení by piloti byli opět schopni služby již tentýž den v 17:25 UTC. Jako poslední problém nastal v rámci doby letové služby. Se začátkem FDP v 7:30 místního času a se čtyřmi sektory není možné všechny lety v rámci jedné FDP uskutečnit.



Maximum possible FDP was exceeded. Limit is: 12:00. Extend max. FDP or shorten your planned FDP by the amount of: 00:50.

Při zadávání vstupních hodnot pro třetí posádku se již vybere možnost prodloužení základní doby letové služby bez možnosti odpočinku za letu. Let ale nemůže začínat podle plánu v 7:30 místního času.

The Rest wasn't sufficient. Do you want to reduce rest period?

Posádka ještě téměř hodinu využívá prodlouženého odpočinku.

První let se může uskutečnit jenom za podmínky letu třetí posádky. První posádka je díky limitu kumulativní FDP neschopna jediného letu delšího 6 hodinám v tomto kalendářním roce. S druhým letem je také problém. Třetí posádka je schopna službu zvládnout, ale s opožděným začátkem o jednu hodinu. Zato druhá posádka by službu mohla podniknout až téměř o týden později z důvodu překročení kumulativní doby služby.

Provozovateli tedy nezbývá nic jiného než posunout jeden z letů na jiný den. Nejvýhodnější nejspíš bude nechat třetí posádku odsloužit obě služby. Nejdřív by se letěla druhá plánovaná služba. Sice s hodinovým zpožděním od plánovaného startu, ale první služba vyžaduje zůstat v Ženevě. Takhle se minimalizují časové ztráty.

Krajním řešením může být provoz první služby třetí posádkou bez zpoždění a provoz druhé služby třetí posádkou za pár dní nebo druhou posádkou až o týden později.

5 Závěr

Problematika sledování letového zatížení posádek je velmi důležitým faktorem bezpečnosti v letectví. Přesto svou komplexností může leckomu zamotat hlavu. A i když si člověk pamatuje všechny podmínky, jsou mnohé kontroly značně zdoluhavé pro ruční výpočty. Jako příklad může posloužit kontrola kumulativní doby letové služby.

Algoritmus přenáší tohle břemeno z člověka na přesně definovaný postup, který svou obecností zaručí, že se nezapomene na žádnou podmínku. Zakomponován do počítačového programu, který poskytuje výpočetní kapacitu, vytváří snadnou pomůcku pro sledování letového zatížení posádek.

Prezentovaná modelová situace má za úkol poukázat na to, jak nesnadný úkol je naplánovat rozpis posádek pro dané služby. Operátor musí dbát na limity v rámci letového zatížení posádek, požadavky zákazníka, dostupnost letounů, a tak dále.

Výsledný produkt diplomové práce je určen primárně pilotům. Aby měli možnost vést si záznamy o svých službách a zároveň si kontrolovat, zda jejich plán služeb není v rozporu s normou letového zatížení posádek. Program mohou využívat i provozovatelé. S přibývajícím počtem pilotů dané společnosti ale klesá efektivita programu a je proto lepší sáhnout po jiném produktu. Obecně se dá říci, že je tento program vhodný spíše pro malé letecké společnosti s počtem zaměstnaných pilotů v řádech jedinců.

Podobnou aplikací, jakou je program diplomové práce je například Flight Duty Calculator z Android obchodu. Ta ale slouží spíše jen pro plánování. Vypočítává maximální možnou dobu letové služby a minimální dobu odpočinku. Nemá žádnou databázi předchozích letů. Další nevýhodou také je, že není aktualizována na nové Nařízení komise (EU) č. 83/2014.

Nejpodobnější diplomovému programu zřejmě je Flight Duty and Rest Calculator, která je dostupný na serveru <https://www.theairlinepilots.com/apps/dutyrest.php>. Je i ve verzi .xlms, tedy soubor Excel s podporou maker. Stejně jako předchozí aplikace ani tato nemá elektronickou databázi. Výsledky svých výpočtů ale určeny pro tisk a následnou archivaci. Jak i v tomto souboru, tak i v diplomovém programu se vstupní data zapisují ve stejném formátu. Podobnost je dána tím, jakým způsobem Excel počítá s časovými konstantami.

6 Bibliografie

- [1] *Zákon č. 49/1997 Sb.: Zákon o civilním letectví*. Česká republika, 1997.
- [2] EASA. *EASA* [online]. Kolín nad Rýnem: Evropská unie, 2017 [cit. 2017-11-25]. Dostupné z: <https://www.easa.europa.eu/>
- [3] EU. *Commission Regulation (EU) No 83/2014*. In: . Brussels: Official Journal of the European Union, 2014.
- [4] *Certification Specifications and Guidance Material for Commercial Air Transport by Aeroplane - Scheduled and Charter Operations CS-FTL.1: Annex to ED Decision 2014/002/R*. In: . EU: EASA, 2014.
- [5] KRČEK, Břetislav a Ivan KOLOMAZNÍK. *Algoritmy a datové struktury* [online]. Ostrava, 2006 [cit. 2017-11-18]. Dostupné z: <http://homen.vsb.cz/~kol70/algoritmy/>. VŠB - Technická univerzita Ostrava.
- [6] PŠENČÍKOVÁ, Jana. *Algoritmizace*. Vyd. 2. Kralice na Hané: Computer Media, 2009. ISBN 978-807-4020-346.
- [7] ČSN ISO 5807. *Zpracování informací.: Dokumentační symboly a konvence pro vývojové diagramy toku dat, programu a systému, síťové diagramy programu a diagramy zdrojů systému*. První. Česká Republika: Český normalizační institut, 1996.
- [8] Algoritmizace a programování. *Matematická biologir* [online]. Brno: Masarykova Univerzita, b.r. [cit. 2018-02-14]. Dostupné z: <http://portal.matematickabiologie.cz/index.php?pg=zaklady-informatiky-pro-biology--algoritmizace-a-programovani--navrh-algoritmu-i--vyvojove-diagramy--vetveni>
- [9] BECHYŇOVÁ, Marta. *Algoritmizace. Stránky k výuce informatiky* [online]. Česká Republika: Gymnázium Vlašim, 2012 [cit. 2018-03-03]. Dostupné z:

<http://www.ivt.mzf.cz/algoritmizace-a-programovani/uvod-do-algoritmu/2-algoritmizace/>

- [10] HALVORSON, Michael. *Microsoft Visual Basic 2008: krok za krokem*. Vyd. 1. Brno: Computer Press, 2008. Krok za krokem (Computer Press). ISBN 978-80-251-2221-1.